ROBUST FRAMEWORK FOR SYSTEM ARCHITECTURE AND HANDOFFS IN WIRELESS AND CELLULAR COMMUNICATION SYSTEMS

A Dissertation

by

VISHAL VINOD VARMA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2008

Major Subject: Electrical Engineering

ROBUST FRAMEWORK FOR SYSTEM ARCHITECTURE AND HANDOFFS IN WIRELESS AND CELLULAR COMMUNICATION SYSTEMS

A Dissertation

by

VISHAL VINOD VARMA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Don R. Halverson
Committee Members,	Deepa Kundur
	Takis Zourntos
	Amarnath Banerjee
Head of Department,	Costas N. Georghiades

December 2008

Major Subject: Electrical Engineering

ABSTRACT

Robust Framework for System Architecture and Handoffs in Wireless and Cellular Communication Systems. (December 2008)

Vishal Vinod Varma, B.Tech., Indian Institute of Technology, Bombay;

M.S., Texas A&M University

Chair of Advisory Committee: Dr. Don Halverson

Robustness of a system has been defined in various ways and a lot of work has been done to model the robustness of a system, but quantifying or measuring robustness has always been very difficult. In this research, we develop a framework for robust system architecture. We consider a system of a linear estimator (multiple tap filter) and then attempt to model the system performance and robustness in a graphical manner, which admits an analysis using the differential geometric concepts. We compare two different perturbation models, namely the gradient with biased perturbations (sub-optimal model) of a surface and the gradient with unbiased perturbations (optimal model), and observe the values to see which of them can alternately be used in the process of understanding or measuring robustness. In this process we have worked on different examples and conducted many simulations to find if there is any consistency in the two models. We propose the study of robustness measures for estimation/prediction in stationary and non-stationary environment using differential geometric tools in conjunction with probability density analysis. Our approach shows that the gradient can be viewed as a random variable and therefore used to generate probability densities, allowing one to draw conclusions regarding the robustness. As an example, one can apply the geometric methodology to the prediction of time varying deterministic data in imperfectly known non-stationary distribution.

We also compare stationary to non-stationary distribution and prove that robustness is reduced by admitting residual non-stationarity.

We then research and develop a robust iterative handoff algorithm, relating generally to methods, devices and systems for reselecting and then handing over a mobile communications device from a first cell to a second cell in a cellular wireless communications system (GPRS, W-CDMA or OFDMA). This algorithm results in significant decrease in amount of power and/or result is a decrease of break in communications during an established voice call or other connection, in the field, thereby outperforming prior art.

To My Parents Sunita and Vinod Varma.

ACKNOWLEDGMENTS

I would like to thank the professors on my committee for their assistance and insight into my research. I would like to give special acknowledgment to Dr. Halverson. I have enjoyed working with him and greatly appreciate the many opportunities he has made available.

Finally, I would like to thank all my friends and family that kept encouraging me all along the tedious process of writing this dissertation and to whom I dedicate this work.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

> Robustness is a central issue in all complex systems. For example, robustness is critical to self-assembling or self-repairing systems, which may be subject to external perturbations, fluctuations, and noise. Other examples of robustness arise in biological and social systems, including questions such as how much diversity is required to sustain an ecosystem, the responses of cellular processes to fluctuations in their environments, the viability of organisms subjected to perturbations in the design of genetic circuitry, and the stability of social organizations in the face of famine, war, or even changes in social policy. Robustness is related to highly optimized tolerance (HOT). Ultimately, an understanding of robustness, both in naturally occurring systems and in engineered systems, will be crucial to what has been called *sustainability*. —Santa Fe Institute

A. History and Prior Art

Engineering has always had to deal with inexact knowledge in both design and analysis. Certain assumptions are made, resulting in so-called optimal algorithms, but the fidelity of such algorithms is limited by the accuracy of the assumptions. Accordingly, often more useful algorithms for the practitioner are those whose performance is less sensitive to the inexact knowledge. This reduced sensitivity is associated with robust procedures [1], and the study of robustness is an important area of research into algorithms, which have the potential to not fail in the field. Algorithms designed for real

---

The journal model is *IEEE Transactions on Information Theory*.

world applications often must rely on inexact knowledge of certain quantities. To be successful, the algorithm must possess a degree of robustness to this inexact knowledge; that is the algorithms performance should not be ultra sensitive to deviations from expected values. Huber Strassen saddle point techniques [2, 3] are often used to overcome such inexact statistical knowledge. However, such techniques are qualitative rather than quantitative in that they provide for stability under the worst conditions, conditions that may never arise in the particular problem under considerations. In recent years, robustness measures have been widely discussed. Many of the techniques used in today's communications, signal processing, and control systems applications rely on techniques for estimating various signal parameters. Furthermore, it has become evident that the degree of robustness associated with the parameter estimator is an important factor that effects overall system performance. Evidence of increased attention given to robustness issues in signal processing applications is given in [1], which is an excellent paper that investigates robustness issues.

In this dissertation we consider the estimation of a system parameter based on data which we allow to be dependent, and either stationary or non-stationary. While an analysis is fairly straight forward for certain cases (e.g., the single step estimation of a data in independent identically distributed -i.i.d.- Gaussian data), the situation becomes far more involved when the data possess statistical distributions which are dependent and non stationary. Hence the motivation to seek procedures that are robust to the inexact statistical knowledge has gathered increased attention.

B. The Huber Strassen Approach

For an algorithm to be successful, it must possess a degree of robustness to the inexact knowledge, i.e., the algorithm's performance should not be too sensitive to

inexact statistical knowledge. Much past work has been performed in engineering robustness research by applications of Huber-Strassen saddle-point techniques [2, 3]. This technique still plays an important role in today's research.

Classically, the noise models were stationary Gaussian with a parametric assumption (all necessary parameters assumed known), but it was quickly realized that some relaxation of such assumptions was necessary. For example, while a Gaussian model might be useful as a first step, it is certainly desirable to allow the entries in its covariance matrix to be imperfectly known and to admit residual non-stationarity.

For these reasons, there has been sustained interest in algorithms that feature robustness. This could be in the context of the algorithm performing well under a Gaussian assumption but with imperfectly known covariance matrix and residual non-stationarity. Classical robustness analysis (e.g. see [4, 5]) has built off the Huber-Strassen saddle-point approach, and many useful results have been obtained.

1. Applied to Estimation Problem [6]

Noise is often modeled with a $\mathcal{N}(0,1)$ distribution. Nevertheless, actual distributions practically never have exactly this distribution. The density function might be characterized as:

$$f(x) = (1-\epsilon)\frac{1}{\sqrt{2\pi}}exp(-\frac{x^2}{2}) + \epsilon h(x),\ x \in \mathcal{R} \tag{1.1}$$

Where h(x) is an arbitrary density, symmetric about zero, with variance

$$\sigma_h^2 = \int_{-\infty}^{\infty}[x^2 h(x)]dx \tag{1.2}$$

finite but not bounded. Then the least squares estimate will have asymptotic variance

$$v_h^2 = \frac{(1-\epsilon)+\epsilon\sigma_h^2}{n\epsilon_\theta} \tag{1.3}$$

Note that v_h^2 can be arbitairly large for any $\epsilon > 0$, since σ_h^2 is not bounded. In particular the worst case asymptotic variance over the class of densities is

$$\sup_h[(1-\epsilon) + \epsilon\sigma_h^2] = \infty \tag{1.4}$$

for any $\epsilon > 0$. This points to a lack of robustness of the least squares estimate for situations in which a small fraction of the noisy samples may come from a high variance distribution. This may happen, for example, in radar communications, in which very high variance impulsive interference may be present in a small fraction ϵ of the measurements. Observations that are improbably large for a given nominal model are sometimes termed outliers. An alternative to asymptotic variance at the nominal model is needed as a design criterion for such situations. One possible way of designing a robust estimator for an uncertainty class $\mathcal{F}$ of noise densities is to seek a function ψ that minimizes the worst case M-estimate variance, $\sup_{f\in\mathcal{F}} V(\psi; f)$. One possible design method is to restrict attention to M-estimates and solve

$$\min_{\psi} \sup_{f\in\mathcal{F}} V(\psi; f) \tag{1.5}$$

The problem has been studied by Huber [2] for general sets $\mathcal{F}$. Within appropriate conditions its solution is basically as follows. Consider the function:

$$I(f) = \int (f')^2/f \tag{1.6}$$

And let f_L be a density in $\mathcal{F}$ that minimizes $I(f)$ over $\mathcal{F}$,

$$I(f_L) = \min_{f\in\mathcal{F}} I(f) \tag{1.7}$$

Then the M estimate with ψ-function $\psi_R(x) = -f_L'(x)/f_L(x)$ solves the equation 1.5. Note that for any f,

$$V(\psi; f)\Big|_{\psi=-f'f} = 1/I(f) \tag{1.8}$$

So that $[ne_\theta I(f)]^{-1}$ is asymptotic variance of the MLE in our model with given f. Thus f_L is the member $\mathcal{F}$ of whose corresponding optimum estimate (the MLE) has the worst optimum performance. For this reason f_L can be considered a least favorable density and the robust M estimate is the best estimate for this least favorable model. The problem $\min_{f\in\mathcal{F}} I(f)$ has been solved for a number of uncertainty models [3] $\mathcal{F}$. For example, for the ϵ-contaminated $\mathcal{N}(0,1)$ model of equation 1.1, the least favorable density is given by

$$f_L(x) = \begin{cases} (1-\epsilon)\frac{1}{\sqrt{2\pi}} exp(-\frac{x^2}{2}) & \text{if } |x| \le k' \\ (1-\epsilon) exp(-k'/2)\frac{1}{\sqrt{2\pi}} exp(-k'(|x|-k')) & \text{if } |x| > k' \end{cases} \tag{1.9}$$

Where k' is a constant given by the solution to

$$(1-\epsilon)^{-1} = 2\Phi k' - 1 + (1/k')(2/\pi)^{1/2} exp(-k'/2) \tag{1.10}$$

The corresponding robust ψ function is

$$\psi_k(x) = \begin{cases} x & \text{if } |x| \le k' \\ k' sgn(x) & \text{if } |x| > k' \end{cases} \tag{1.11}$$

Thus, as in the analogous hypothesis-testing problem, robustness is brought about by limiting the effects of outliers.

2. Applied to Detection Problem

In this example [6], for an i.i.d. sequence $Y_1, ..., Y_n$, with densities p_0 and p_1, and marginal distributions P_0 and P_1, an optimum test is based on the following ratio.

$$L(y) = \prod_{k=1}^{n} \frac{p_1(y_k)}{p_0(y_k)} \tag{1.12}$$

In this problem of testing between two marginal densities, it is often the case that both are imperfectly unknown. If the density of uncertainty is known or can be assumed, we might model this uncertainty as

$$(1-\epsilon)P_j + \epsilon M_j, j = 0, 1 \tag{1.13}$$

Where are M_0 and M_1 are the unknown contaminating distributions and ϵ as a value between 0 and 1 is the degree of uncertainty, since the condition where p_1 is much greater than p_0 occurs most often under H_1, an unknown distribution M_0 could have for instance all of its probability just where is p_1 much greater than p_0. This would generate more errors under H_0 of the order of $1-(1-\epsilon)^n$. Detectors are often desired to operate at the false alarm probabilities of $10^{-4} - 10^{-5}$. The opposite result can occur when p_1 is much less than p_0 in which case the error results from an already low $L(y)$ being driven below the threshold. Thus in the presence of noise for which a complete statistical description of data is unavailable the output cannot be considered robust unless the ratio $\frac{p_1(y_k)}{p_0(y_k)}$ is bounded away from zero below and away from large values above. To prevent an increase in the false alarm rate, the response curve of can be truncated as follows

$$\Big[l\Big]_a^b k(y) = \begin{cases} b & \text{if } l(y) > b \\ l(y) & \text{if } a \leq l(y) \leq b \\ a & \text{if } l(y) \leq a \end{cases} \tag{1.14}$$

For correctly chosen a and b, the above is more robust, however information is obviously lost. Moreover the problem at hand is calculation of the values a and b. Although there are various schemes for achieving a and b, the most robust scheme is that of Huber which calculates a and b based on achieving the best performance for the worst case distribution for the noise. The solution to this problem is called the saddle point solution and is the highest point on the lowest curve of a saddle comprised of all such curves calculated from the set of possible distributions indicated by a particular nominal.

A Huber type saddle point methodology essentially makes the best of a worst-case scenario. It is often the case that the noise distribution, characterized as a function F, not precisely known. This actual distribution, F, is assumed to belong to a set of distributions that with equal likelihood could possibly characterize the noise. Of this set the worst possible distribution is chosen and for this distribution one attempts to find the best possible estimator. Thus Huber-Strassen approach seeks the optimum (in terms of estimators) lower bound (in terms of worst possible distribution of noise). Known also as the minimax theory of robust estimation, this method does not take into consideration the likelihood or possibility of occurance, of the worst possible distribution, assuming all possible distributions are not of the same likelihood. If for instance the probability of occurance of distributions producing high error including the worst possible distribution is very low, one might choose a more favorable distribution for the sake of performance that which is worst, sacrificing but little robustness. Such a problem would then have built in a certain robustness to a more optimistic choice of distribution. Without the benefit of such knowledge the Huber approach operates under the assumption that such choice for the sake of performance would be catastrophic.

In a one step predictor, prediction is accomplished by a regression to the means

by a factor k found in the solution of the least mean squares problem. The result of Huber′s method is a simple truncation at a point. Where this truncation occurs is found in the solution to a minimax type problem. While the solution to this problem may be difficult, the action is a simple censuring of the output beyond a certain value.

It has been shown [7] that this truncation of the output is optimal and that no better result is obtainable under these conditions in the least mean square sense. This is with the assumptions that all possible distributions are equally likely and that for the signal detection problem at hand error in the least mean square sense in the only consideration when accessing performance.

Significantly, the Huber-Strassen approach relies on the existence of known and well understood distributions, P_j, such that they closely fit the distributions that may be found in nature. If this is not the case, ϵ must be large, so the influence of M_j will also be large. M_j itself is an unknown distribution that must be found to compensate for the lack of fit by P_j.

Thus, we need P_j to be less than the given distribution. But nature does not always present a convenient distributions and it may be found that to meet these criteria, ϵ must be large and in fact greater than $1/2$. Even so calculation of the Huber Strassen censuring point is unwieldy at best and may yield a solution so unnecessarily safe as to be useless. A case in point is a distribution presented by nature that is not easily fit by any unknown and well understood distribution. The possibility exists that a fit by any known distribution might be so poor that, ϵ approaches 1. Such a scenario would yield a censoring point near zero. In other words, we would be reduced in such a case to choosing the mean. With the Huber method one must specify not only the ϵ-contamination, but the distribution H as well. Neither quantity is well defined. It seems that if one specifies as underlying distribution F which is well known and as close as possible to the actual, then the goal is to specify a contaminating distribution,

H, that can be dialed in to the amount ϵ to produce the distribution presented by nature, F_N. Once this is obtained, can be expressed as,

$$F_N = (1 - \epsilon)F + \epsilon H \tag{1.15}$$

The worst case H must be found. Furthermore the choice of ϵ is *ad hoc*.

C. Limitations of the Huber-Strassen Approach

Although the Huber Strassen approach has been proven to be optimal in a mean square sense, changes in physical conditions may limit or skew changes in variance or correlation coefficients in a particular direction. Thus, the change of generations the high degree of error that is assumed by the Huber-Strassen approach may become remote, or, even more possible. This in turn can invalidate the results of a Huber Strassen censuring approach towards robustness. In such situations a censuring approach may prove to be overly cautious or not cautious enough. As we have seen the Huber-Strassen method makes use of ad hoc selection of a critical factor, and in this way we find it insensitive to naturally occurring distributions that may be very difficult to characterize. The Huber-Strassen method is not naturally quantitative; one obtains an algorithm as a solution of a saddle point equation which is by definition: "the" robust solution. Algorithms which are very close to the solution are simply not robust, and there is not a natural measure of how close they are to being robust. Secondly, the adaptation by practitioners in such fields as engineering are bound by Huber's concept of robustness as a subject of interest to statisticians who focused on outliers. For many practitioners, this may not be what they mean by robustness, since outliers might not be the chief problem. Third, the saddle point method restricts the type of uncertainty admitted through canonical models such as ϵ-contamination, and

finally, the method resists admitting non-stationarity and dependency.

D. The Differential Geometry Approach

Much past work has been performed in engineering robustness research by applications of Huber-Strassen saddlepoint techniques [2, 3]. Even though this technique still plays an important role in today's research, it contains major limitations which have inspired an alternative approach using differential geometric tools. This new direction of research [8]-[18], allows engineers to combine both performance and robustness into the development of an algorithm. An alternative to the classical Huber-Strassen saddle point approach has been undertaken by Halverson and collaborators using a differential geometry approach. This approach admits the development of natural quantitative measures of the degree of robustness, allowing the designer to incorporate both performance and robustness into the development of an algorithm. This work is naturally quantitative and views the focus of robustness not on outliers but on perturbations away from a nominal. This may be of more interest to practitioners in fields such as engineering, where one first often makes a "seat of the pants" guess (i.e., to choose nominal values) and requires an algorithm that will tolerate an imperfect guess. In addition, the work readily admits non-stationarity and dependency. The general idea of this work is that the imperfectly known quantity (parameter vector or distribution) is allowed to vary about its nominal over a differentiable manifold which models a local "neighborhood" about the nominal. As the variation occurs, the performance (e.g., false alarm rate or detection probability for detectors, mean L_p error for estimators) changes. The greater the change: the less robustness; thus gradient provides a convenient measure. Early work [8, 9, 10, 11] focused on local robustness very close to the nominal, and so gradient at the nominal was a key ele-

ment in robustness measure. Later work [12, 13] allowed for non-local neighborhoods, with [13] even allowing the robustness to be computed as a hybrid between the local and the non-local concepts, thus admitting an emphasis on outliers *a la Huber* if desired. While gradient was commonly employed, there was also consideration to context where gradient was insufficient, and the second-order measure offered by Gaussian curvature was applied [15, 16, 17]. All of this later work features Euclidian models for the parameter manifold. Recent work [15, 18] has explored the admission of non-Euclidian models. The reason for doing so is not to do more complex mathematics for its own sake, but to better model reality. For example, if a covariance matrix is imperfectly known, one approach would be to model this by simply allowing each entry to vary by, say, $\pm\delta$. If the nominal matrix is positive definite, then for sufficiently small δ the actual matrix will stay positive definite. But how small? This Euclidian model thus puts limits, possibly severe, on how much variation can be considered. But, a practitioner might need to take into account large variations. A Euclidian model will thus fail: it will admit matrices which are not positive definite and thus cannot be potential covariance matrices. The best approach would be to *impose* positive definiteness *a priori*, resulting in a non-Euclidian structure. Naturally, employing a non-Euclidian Riemannian manifold complicates the mathematics, but it provides a much more versatile tool to address the needs of the user. It is also a feasible and tractable approach; for example, in [15, 17] both gradient and curvature were employed to measure robustness for linear estimation using non-Euclidian model with a "biased" perturbation interpretation. In [18] the same general problem was expanded using gradient to include an "unbiased" perturbation interpretation, and this was compared to the "biased." Our proposed work will, for the first time, direct the non-Euclidian model toward multiple tap prediction filter and non stationary estimation theory.

E. Applications

Estimation and prediction theory can be found at the center of many electronic signal processing systems designed to extract *information*. The theory of estimation, originally developed within the area of statistics, is applied today in many different fields. A non-exhaustive list of the systems that incorporate this purpose is given as follows: radar, speech, sonar, communications, seismology, image analysis, econometric forecasting, etc..

All of these applications share the same common function of needing to estimate the values of a group of parameters. For example, within a radar system one might be interested in determining the position of an aircraft. To accomplish this, the transmission of an electromagnetic pulse is reflected off of the aircraft causing the antenna to receive an echo a few moments later. Clearly, if the the round trip delay can be measured, one can easily compute the distance between the plane and the antenna, even though the echo is decreased in amplitude due to the propagation losses and the corruption of the signal induced by the channel.

Another example is speech processing systems, such as speech recognition. This system corresponds to the recognition of speech by a device like a computer. One of the many tasks the computer can execute is trying to recognize speech sounds like vowels or consonants. In order to do so, a computer tries to compare the spoken vowel with waveforms stored in memory and then chooses the closest one to the spoken vowel, the one that minimizes the *distance measure*. The main difficulty here is when the pitch of the speakers's voice differs from the voice recorded during the training session. Due to this problem inherent to the wave forms, one should try to choose different attributes that are less susceptible to variations. For example the spectral envelope will not change with the pitch since the Fourier transform of a

periodic signal is a sampled version of the Fourier transform of one of the period of the signal. The period only affects the spacing between frequency samples, not the values.

In all of these systems, one faces the problem of the *extraction* of the parameter values based on *continuous* waveforms. The same problem arises in the use of digital computer. The equivalent problem is to extract different parameter values from a *discrete-time* waveform or a data set. Mathematically, we now have a N-point data set $\{x[0], x[1], ..., x[N-1]\}$ which depends on an unknown parameter θ.

If one wishes to determine θ based on the data set or to define an *estimator*, he will obtain:

$$\widehat{\theta} = g(x[0], x[1], ..., x[N-1]) \tag{1.16}$$

where g is some function.

One of the first people to address this kind of problem was Gauss in 1795 with the use of least squares data analysis to predict the movement of the planets (see [19]).

Detection theory is an area of classical interest. Simply put, it employs classical hypothesis testing to make a "signal with noise" or "noise only" decision. The signal can be random or not, but the noise is modeled as a random process. The "signal and noise" are not limited to electro-magnetic phenomena, but can be interpreted as economic or medical trends – we are really referring simply to a choice between two hypotheses. Applications of classical interest include sonar (naval warfare; oil exploration); radar (military; remote sensing) and more recently, radio astronomy.

F. Differential Information Geometry to Robustness

The aforementioned research employs geometric tools which have been designed specifically with robustness in mind, and we show that these tools lead to success in our proposed applications. But it can also be enlightening to employ more than one perspective, and if multiple perspectives achieve some degree of consistency then the relevance of the results to the practical domain can be enhanced. For example Fourier series analysis tells us that periodic signals are composed of harmonic components whose importance is measured by the respective Fourier coefficients. A Fourier series is specifically employed with a periodic signal or a signal with periodic extension in mind. On the other hand, Fourier transform is applied most commonly with a non periodic signal in mind, yet if we apply this to the periodic domain the same qualitative result is obtained harmonic decomposition with weighting proportional to the Fourier series coefficients. This consistency strengthens the philosophical position that such a view of periodic signals is a metaphysical absolute and not artifact of definition. There exists a body of work (see for example, [20]-[33]) which applies differential geometry to a variety of areas of statistical interest, in particular, estimation and hypotheses testing. The methods are quite general and at first glance rather abstract, but they have the potential for providing an interpretation of robustness. To quote from [21] "information geometry provides mathematical science with a new framework for analysis. This framework is relevant to a wide variety of domains. It has already been usefully applied to several of these, providing them with a new perspective from which to view the structure of the systems which they investigate. Nevertheless, the development of the field of information geometry can only [sic] be said to have begun". We have investigated whether or not these methods can be easily applied to robustness. Because the techniques are very mathematical, we believe

it involves building from the ground up– and that would be more than a dissertation in itself. Thus, to exhaustively pursue such an application is beyond the scope of this dissertation. Perhaps this future work would show, the techniques naturally connect with robustness, seek such a nexus and to either develop it or to determine that it is unlikely that one exists. While not essential, success in this future effort could provide a useful confirmation of the efficacy of our approach.

G. Purpose and Overview of This Dissertation

The thrust of the work proposed here is to extend the differential geometric approach toward robustness in several important ways. First, we note that past work made use of both Euclidean space [8, 9, 13, 16] and curved space [15, 18] to model the inexact knowledge. Nevertheless, this past work has admitted non stationary data only for Euclidean space [8]. Since it is very likely that the real world will feature various degrees of non- stationarity, and since we have seen in that often a curved space model [15, 18] is more appropriate, we applied the differential geometric approach to measure robustness in the scenario where the data is non stationary with a curved space model. We compared these results with the robustness values obtained under a non-stationary assumption to see if residual stationarity degrades robustness. For example, in [8] it was found that robustness for Euclidean models could be cut approximately in half by residual stationarity. Finally we have gone beyond the Halverson differential geometric approach to research information geometry [20]-[33] and investigate applications toward robustness.

The goals of this dissertation include:

- Develop a framework for robust system architecture.
- To illustrate how non-Euclidean geometry can be applied toward measuring

robustness of systems of engineering interest. While we specifically illustrate an application for estimation with a multiple tap filter, our methods will be seen to have broad potential application

- To compare the biased versus unbiased approach of [18] to see if they yield similar results. If so, then the biased approach might be appealing because it is easier to compute

- To compare the robustness of the estimator for stationary data versus non-stationary. Is there a loss of robustness due to non-stationarity?

- To develop a Robust iterative handoff Algorithm

- To go beyond the Halverson differential geometric approach to research information geometry and investigate feasibility of nexus toward robustness.

Consider a performance function $P : R^m \rightarrow R$. For example, the covariance matrix entries may lie in a subset of R^m, and P might be the least squares error of a linear estimator employed with an imperfectly known variance matrix. Using the techniques described in [8, 9, 10, 11], one can easily associate the robustness of an algorithm with changes in P as one moves away from the nominal point in R^m. The most convenient way to do this locally is to look at the rate of change of P as one moves in the manifold M spanned by the imperfectly known quantities. While potentially misleading, the use of slope is convenient and has been found to be a good indicator of the actual performance in many situations where the more sensitive additional measure provided by curvature [15, 16, 17] is not necessary.

As an example of the application of the above approach, consider applications which involve Gaussian-distributed data. The Gaussian covariance matrix is crucial to an analysis of such an application, but it is unrealistic to assume we will know

the matrix perfectly. The set of possible entries for the matrix is imperfectly known but its constraint, such as being positive definite, might be known. This and other possible constraints will result in the entries lying in a manifold which may very well be non-Euclidian. More detail of this proposed work is presented in chapter III.

We remark that the work in chapter II will be dedicated to estimation, prediction and detection theory and will also include discussion of applications, historical context, assumptions, and will provide more detailed comments of some of the elements of this introduction. Chapter III will employ gradient to generate a quantitative robustness measure and will also be based on data estimation. Chapter III will therefore be dedicated to the applications and tools of differential geometry to robustness while Chapter IV will be dedicated to the extension of the work using non-stationary distribution. Chapter V allows one to observe the effect of stationarity on robustness and will be dedicated to generate a comparative quantitative robustness measure based on signal estimation. Chapter VI presents robust handoff algorithm developed during the internship opportunity at Qualcomm. Finally, chapter VII will present a complete summary of all the results obtained along this dissertation and discuss potential future work.

CHAPTER II

TOOLS, FUNDAMENTALS AND BACKGROUND

This chapter presents two main tasks in signal processing, which are detection and estimation of a signal leading to the making of a decision. It clearly shows how these two tasks are closely related and then moves on to make an in-depth presentation of the tools needed to form differential geometry for this research.

A. Estimation Theory in Signal Processing

In modern life, estimation theory can be found at the center of many electronic signal processing systems designed to extract *information*. The theory of estimation, originally developed within the area of statistics, is applied today in many different fields. A non-exhaustive list of the systems that incorporate this purpose is given as follows: radar, speech, sonar, communications, seismology, image analysis, etc..

All of these applications share the same common function of needing to estimate the values of a group of parameters. For example, within a radar system one might be interested in determining the position of an aircraft. To accomplish this, the transmission of an electro-magnetic pulse is reflected off of the aircraft causing the antenna to receive an echo a few moments later. Clearly, if the the round trip delay can be measured, one can easily compute the distance between the plane and the antenna, even though the echo is decreased in amplitude due to the propagation losses and the corruption of the signal induced by the channel.

Another example is speech processing systems, such as speech recognition. This systems corresponds to the recognition of speech by a device like a computer. One of the many tasks the computer can execute is trying to recognize speech sounds like vowels or consonants. In order to do so, a computer tries to compare the spoken

vowel with waveforms stored in memory and then chooses the closest one to the spoken vowel, the one that minimizes the *distance measure*. The main difficulty here is when the pitch of the speakers's voice differs from the voice recorded during the training session.

Due to this problem inherent to the wave forms, one should try to choose different attributes that are less susceptible to variations. For example the spectral envelope will not change with the pitch since the Fourier transform of a periodic signal is a sampled version of the Fourier transform of one of the period of the signal. The period only affects the spacing between frequency samples, not the values.

In all of these systems, one faces the problem of the *extraction* of the parameter values based on *continuous* waveforms. The same problem arises in the use of digital computer. The equivalent problem is to extract different parameter values from a *discrete-time* waveform or a data set. Mathematically, we now have a N-point data set $\{x[0], x[1], ..., x[N-1]\}$ which depends on an unknown parameter θ.

If one wishes to determine θ based on the data set or to define an *estimator*, he will obtain:

$$\widehat{\theta} = g(x[0], x[1], ..., x[N-1]) \tag{2.1}$$

where g is some function.

One of the first people to address this kind of problem was Gauss in 1795 with the use of least squares data analysis to predict the movement of the planets (see [19]).

1. The Mathematical Estimation Problem

In order to determine good estimators the first step is to mathematically model the data. The data being random, we can describe it by its *probability density function*

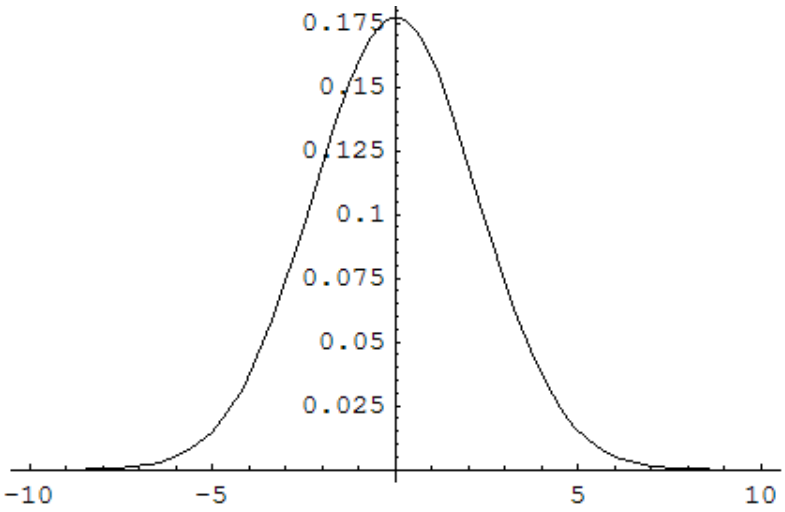

Fig. 1. Gaussian distribution with zero mean

(PDF). As such, the performance of the estimator can only be completely described *statistically*. The unknown parameter θ *parameterized* the PDF, i.e., within the class of PDFs each PDF is different due to the different value of θ.

For example, if N=1 and θ denotes the mean of a Gaussian random variable, then the PDF might be:

$$p(x[0];\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} exp[-\frac{1}{2\sigma^2}(x[0]-\theta)^2] \tag{2.2}$$

which is shown in figure 1 for $\theta = 0$ and figure 2 for $\theta = 5$. The x-axis corresponds to x[0] and the y-axis corresponds to $p(x[0];\theta)$.

It is easy to realize that the value of the mean affects the probability of x[0], and consequently the specification of the PDF is critical in determining a good estimator.

In real life problems, we are not given a PDF but instead we have to choose one that is not only consistent with the problem constraints and any prior knowledge, but one that is also mathematically tractable.

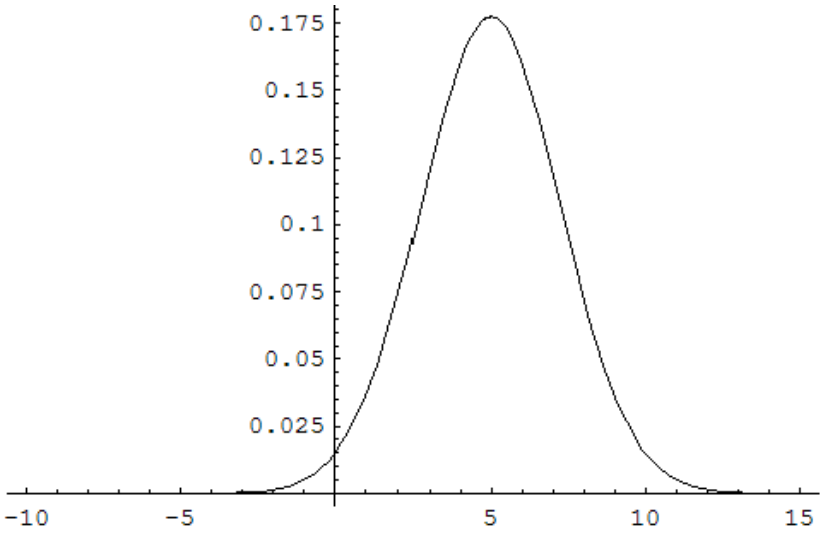

Fig. 2. Gaussian distribution with $\mu = 5$

2. Example

As an example, consider the DC level in White Gaussian Noise (WGN). Let's consider the observations: x[n]=A+w[n] where n=0, 1,...,N-1, where A is the parameter to be estimated. A reasonable model for w[n] is WGN or each sample of w[n] has the PDF $N \sim (0, \sigma^2)$ which denotes a Gaussian distribution with a mean of 0 and a variance of σ^2 and is uncorrelated with the other samples. Thus, the PDF becomes:

$$p(x[n];\theta) = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} exp[-\frac{1}{2\sigma^2}\sum_{n=0}^{N-1}(x[n]-A)^2] \tag{2.3}$$

Therefore a reasonable estimator for the average value of x[n] is:

$$\widehat{A} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] \tag{2.4}$$

The performance of the estimator is critically dependent on the PDF assumptions. One can only hope that the estimator is *robust*, in that slight changes in the PDF do not severely affect the performance of the estimator.

An estimation based on PDFs is described as *classical* estimation because the parameters of interest are unknown but *deterministic*. When one incorporates the prior knowledge of the PDF, therefore assuming that the parameter is no longer deterministic, such an approach is called *Bayesian* estimation. The parameter we are attempting to to estimate is then viewed as a *realization* of the random variable θ. The data are now described by the *joint* PDF:

$$p(x;\theta) = p(x|\theta)p(\theta) \tag{2.5}$$

where $p(\theta)$ is the prior PDF, summarizing the knowledge about θ before any data are observed, and $p(x|\theta)$ is the conditional PDF, summarizing our knowledge provided by the data x conditioned on knowing θ.

Once the PDF has been specified, the problem is to determine an optimal estimator or a function of the data, knowing that the estimator may depend on other parameters. An estimator may be thought of as a *rule* that assigns a value to θ for each realization of x. The *estimate* of θ is the value of θ obtained for a *given* realization of x (see [35]).

B. Detection Theory in Signal Processing

Modern *detection theory* is fundamental to the design of electronic signal processing systems for *decision making* and the extraction of the information. Detection theory is mostly employed in weak signal and/or high noise situations where a decision must be made as to whether a noise-corrupted signal is present [36].

The making of the decision is based on the extraction of the information out of the available data [36]. The mathematical techniques needed to extract as much information from the data will provide the necessary interface between the need to

make a detection decision and the desire to optimize the accuracy of the decision. Ultimately the goal is to use the data as efficiently and accurately as possible.

Most of the results referenced above deal with the case where the signals are deterministic or of known form with random parameters. There are, however, applications in which non-homogeneous propagation media or incomplete knowledge of the signal source invalidates such an assumption regarding the signal.

In these cases, the signal would be best modeled as a random process. There has been a great amount of effort expended in this area; for example, the Gaussian signal case has been considered in many publications like [36], as has the detection of a random signal in Gaussian noise.

1. The Detection Problem

One of the simplest detection problems is to decide whether a signal embedded in noise is present, or if only the noise is present. An example of this problem would be the detection of a plane based on the signal received by a radar. This type of problem is called a *binary hypothesis testing problem*, since one needs to decide between two possible outcomes, signal plus noise present versus just noise present.

$$\begin{aligned} \mathcal{H}_0 &: \textit{signal not present} \\ \mathcal{H}_1 &: \textit{signal present} \end{aligned}$$

The detection decision then reduces to the choice of $\mathcal{H}_0$ and $\mathcal{H}_1$. In order to decide between $\mathcal{H}_0$ and $\mathcal{H}_1$, we have to rely on the available data, which may be collected either continuously or discretely. In the discrete time case, the decision is

between:

$$\begin{aligned} \mathcal{H}_0 &: F_0(.,...,.) \\ \mathcal{H}_1 &: F_1(.,...,.) \end{aligned}$$

where the indicated point distributions F_0 and F_1 apply to n samples and traditionally correspond respectively to "noise only" and "signal with noise" situations.

Further discussion of the decision theoretic background for continuous time detection theory as applied to radar detection in Gaussian noise appears in literature such as [36]. One should note that the random process involved in the previous discussion has historically been modeled as Gaussian. However, much of the noise encountered in real-life situations is highly non-Gaussian. In these situations, continuous time detection often becomes mathematically intractable. Because of this, and also due to the increasing use of discrete time systems, we will limit the remainder of our consideration of detection theory to discrete time case.

Let's consider a case where the noise is deterministic. We will further limit consideration to the common case where the signal is additive to the noise. The detection problem then becomes:

$$\begin{aligned} \mathcal{H}_0 &: F_0(.,...,.) \\ \mathcal{H}_1 &: F_1(.,...,.) \end{aligned} \tag{2.6}$$

where F_0 corresponds to n noise samples and F_1 corresponds to each of these samples respectively added to the i-th signal value s_i. We observe realizations $\{y_i\}_{i=1}^n$ of the observation random process $\{Y_i\}_{i=1}^n$ and the s_i are known constants.

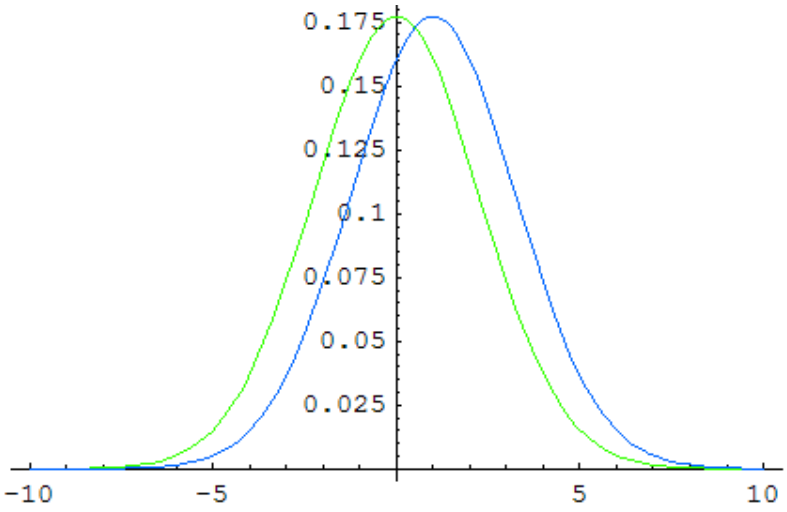

Fig. 3. Hypothesis testing: PDF of x[n] for signal present and signal absent

2. The Mathematical Detection Problem

The model of the detection problem has a form that will allow us to apply the theory of *statistical hypothesis testing* (see figure 3). As we did previously, one can consider, as an example, the detection of a DC level of amplitude A=1 embedded in WGN w[n] with variance σ^2. In order to simplify the discussion one can assume that we base the decision on the only sample available. Hence, one effectively needs to decide between the hypotheses x[0]=w[0] (noise) and x[0]=A+w[0] (signal plus noise). Since the main assumption is to consider the noise to be zero mean, one will need to decide that the signal is present if x[0]>1/2 and if noise only is present x[0]<1/2 -since $E\left[x[0]\right] = 0$ if only noise is present and $E\left[x[0]\right] = A$ if noise and signal are present-.

Clearly there will be an error whenever a signal is present and w[0]<1/2 or whenever noise only is present and w[0]>1/2. We cannot expect to make the correct decision all the time, but hopefully we can maximize the number of times that we make a good decision. The performance of any detector will depend upon how different the PDFs of x[0] are under each hypothesis. Multiple studies have already proven that

the detection performance improves as the "distance" between the PDFs increases or as A^2/σ^2, also called the signal-to-noise ratio (SNR), increases (see [36]).

This study illustrates the basic result that the detection performance depends on the *discrimination* between the two hypotheses or equivalently between the two PDFs. More formally, we model the detection problem as one of choosing between $\mathcal{H}_0$, which is termed the noise-only hypothesis, and $\mathcal{H}_1$, which is the signal present hypothesis.

The PDFs under each hypothesis are denoted by $p(x[0];\mathcal{H}_0)$ and $p(x[0];\mathcal{H}_1)$, which are for example:

$$p(x[0];\mathcal{H}_0) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}x[0]^2) \tag{2.7}$$

and

$$p(x[0];\mathcal{H}_1) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(x[0]-A)^2) \tag{2.8}$$

Ultimately, one will need to decide between the two PDFs as to which one is going to minimize the probability of making an error or equivalently maximize the probability of making a right decision.

Most of the results referenced above deal with the case where the signals are deterministic or of known form with random parameters. There are, however, applications in which non-homogeneous propagation media or incomplete knowledge of the signal source invalidates such an assumption regarding the signal. Such applications arise, for example, in sonar detection, radio astronomy, and seismic exploration.

In this cases, the signal would be best modeled as a random process. There has been a great amount of effort expended in this area; for example, the Gaussian signal case has been considered in many publications like [37, 38, 39], as has the detection of a random signal in Gaussian noise.

3. Detector Fidelity

In order to formalize the notion of fidelity of a detector, it has been found useful to employ two quantities, denoted by α and β, and defined by:

$$\begin{aligned} \alpha &= \textit{probability of choosing } \mathcal{H}_1 \textit{ when } \mathcal{H}_0 \textit{ is true.} \\ \beta &= \textit{probability of choosing } \mathcal{H}_1 \textit{ when } \mathcal{H}_1 \textit{ is true.} \end{aligned} \tag{2.9}$$

Classically, α has been called the "false alarm probability," and β the "detection probability". Ideally, one would like a detector to have the lowest α possible and the highest β possible. The problem is that there is a tradeoff between the two.

For example, if one designs a detector in an effort to make α small, it will bias the decision more toward $\mathcal{H}_0$ than for a detector designed to operate at a higher α level, with the result that β is reduced.

In view of this, there are several approaches which may be taken to formalize the notion of fidelity of a detector. For example, one could employ the *probability of error criterion* (minimize the probability of error) [40], the *Bayes criterion* (minimize the "average risk" associated with a weighted cost formulation) [41], the *minimax criterion* (minimize the maximum risk) [42], or finally the *Neyman-Pearson criterion* (constrain α and maximize β) [43].

One can easily remark that detectors designed under the first three criteria still possess an associated α and β; it can also be shown that many detectors reduce to the form of a Neyman-Pearson detector [44]. Also, α is usually associated with some sort of cost, and thus it is often desirable to constrain α to be no greater than some small value (e.g.$\alpha \leq 0.05$ or 5%). For this reason, the Neyman-Pearson criterion is especially popular and will be of immediate interest here.

The next step is to, when possible, design a detector with a maximal β for a

constrained α. In view of the Neyman-Pearson lemma [45], the optimal detector takes the following form:

$$\Lambda(y_1, ..., y_n) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} T \tag{2.10}$$

where Λ is a function of the observations and T is a deterministic constant. That is, the optimal detector chooses $\mathcal{H}_1$ if $\Lambda(y_1, ..., y_n) > T$, and chooses $\mathcal{H}_0$ if $\Lambda(y_1, ..., y_n) < T$. The Neyman-Pearson also gives the form of Λ. If P_i denotes the *probability measure* induced by the $Y_1, ..., Y_n$ under $\mathcal{H}_1$, and μ is the *dominating measure*, then

$$\Lambda = \frac{dP_1}{d\mu} / \frac{dP_0}{d\mu} \tag{2.11}$$

In this case the joint densities $P_i(y_1, ..., y_n)$ of $Y_1, ..., Y_n$ under $\mathcal{H}_i$ exist, then we have the more recognizable form.

$$\Lambda(y_1, ..., y_n) = \frac{P_1(y_1, ..., y_n)}{P_0(y_1, ..., y_n)} \tag{2.12}$$

This form is called the *likelihood function*. In certain cases, the form of the Neyman-Pearson detector simplifies. For example, if the noise process is first order stationary and "white", we then have, since $p_1(y_1, ..., y_n) = p_0(y_1 - s_1, ..., y_n - s_n)$,

$$\Lambda(y_1, ..., y_n) = \frac{p_0(y_1 - s_1)p_0(y_2 - s_2)...p_0(y_n - s_n)}{p_0(y_1)p_0(y_2)...p_0(y_n)} \tag{2.13}$$

where $p_0(.)$ denotes the univariate density of N_1. If we compare the above to the threshold value T, and take a natural logarithm of each side of the inequality, the Neyman-Pearson test becomes:

$$\sum_{i=1}^{n} \Big[ln\Big(p_0(y_i - s_i)\Big) - ln\Big(p_0(y_i)\Big)\Big] \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \widehat{T} \tag{2.14}$$

where $\widehat{T} = \ln T$. The optimal detector, which is now called the *log-likelihood* ratio, reduces to figure 4.

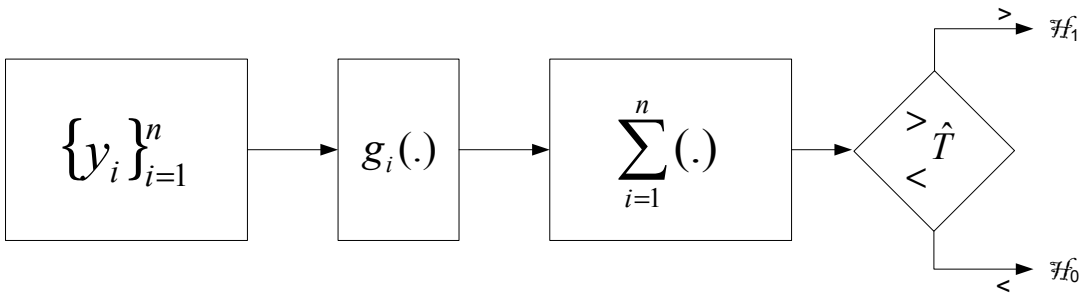

Fig. 4. Linear detector equivalent to a matched filter

where g_i is a time varying zero memory nonlinearity. If the noise is Gaussian, the nonlinearities g_i become linear, yielding a matched filter as in the continuous time case. Also, if the s_i are equal (the "sure" signal case), then the nonlinearity is time invariant.

If the noise is nonwhite, the Neyman-Pearson detector unfortunately is not of such a simple form as above. Because of modern, high speed sampling, dependency between the samples is often no longer negligible, and in these cases a Neyman-Pearson approach to detector design often becomes intractable because of inexact knowledge of the required n-th order densities of the noise, as well as inability to determine the statistics of the likelihood ratio Λ.

C. Introduction to Differential Geometry and Relativity

We wish to remind the reader that differential geometry is a very abstract and challenging field. For a full understanding of those concepts, one should refer himself to the proper mathematical references (see [46] for example). However, in the interest of acquainting the interest of the reader, we now present a short heuristic description of differential geometry.

Differential Geometry is the language of modern physics as well as an area of

mathematical delight. Typically, one considers sets which are manifolds (that is, locally resemble Euclidean space) and which come equipped with a measure of distances. In particular, this includes classical studies of the curvature of curves and surfaces.

In mathematics, differential topology is the field dealing with differentiable functions on differentiable manifolds. It arises naturally from the study of the theory of differential equations. Differential geometry is the study of geometry using calculus. Together they make up the geometric theory of differentiable manifolds - which can also be studied directly from the point of view of dynamical systems.

1. Preliminaries: Distance, Open Sets, Parametric Surfaces and Smooth Function

In order to be able to speak about smooth manifolds, a review of the topology is necessary. A *n-dimensional Euclidian* space corresponds to:

$$E_n = (y_1, y_2, ...y_n | y_i \in \mathcal{R})$$

where R is the set of real numbers. Thus, E_1 represents just a line, where E_2 represents an Euclidian plane, and E_3 represents a 3-dimensional Euclidian space.

The *norm*, also called "magnitude", $\|y\|$ of $y = (y_1, y_2, ..., y_n)$ in E_n is defined to be:

$$\|y\| = \sqrt{y_1^1 + y_2^2 + ... + y_n^2}$$

which can be viewed as its distance from the origin. Therefore the distance between two points , y and z, will be:

$$z = \|z - y\| = \sqrt{(z_1 - y_1)^2 + (z_2 - y_2)^2 + ... + (z_n - y_n)^2}$$

A subset U of E_n is called *open* if, for every y in U, all points of E_n within some

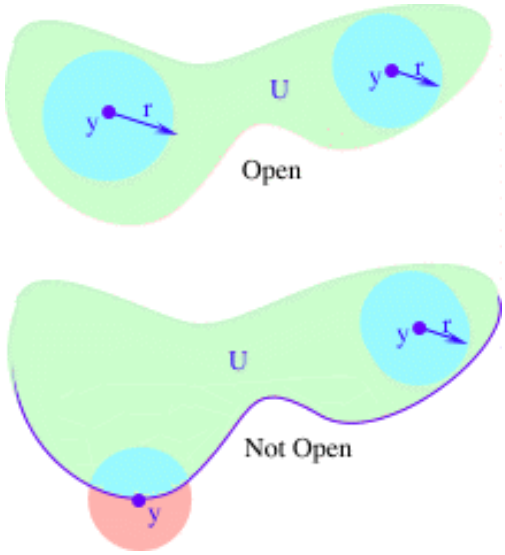

Fig. 5. Representation of open and not open sets

positive distance r of y are also in U. In figure 5, the top set U is a set that does not include any boundary points and is therefore *open*. Thus, given any point y in U, we can find a little ball centered at y that lies entirely within U.

The lower set is not open since it includes on its lower part some boundary points. For example, if one chooses y on the boundary, then it is impossible to find a little ball centered at y entirely contained within U.

Intuitively, one can visualize an open set as a solid region minus its boundary. If the boundary is included then we will get a *closed set*. The closed set is formally defined as the complement of the open set.

Let $M \subset E_s$. A subset $U \subset C$ is called open in M if, for every y in U, all points within some positive distance r of y are also in U. For the parametric paths and surfaces in E_3, from now on, the three coordinates of 3-space will be referred to as $y_1,\ y_2,\ and\ y_3$ [47].

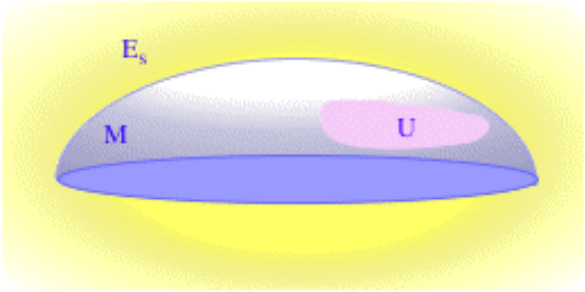

Fig. 6. Open set in M

In figure 6, M is the hemisphere, E_s is 3-dimensional space and U is a small "patch" on M that excludes its boundary. Thus U is not open in E_s, since there are points in E_s arbitrarily close to U that lie outside U. However, it is open in M, since given any point y in U, all points of M within a small enough distance from y are still in U.

This will help define a smooth path in E_3 as a set of three smooth (infinitely differentiable) real-valued functions of a single real variable t:

$$y_1 = y_1(t),\ y_2 = y_2(t),\ \textit{and}\ y_3 = y_3(t)$$

The *parameter* of the curve is the term that usually describes the variable t. The path is *non-singular* if the vector $(\frac{dy_1}{dt},\ \frac{dy_2}{dt},\ \ \frac{dy_3}{dt})$ is nowhere zero.

A smooth path in E_n is defined as a collection of smooth functions $y_i = y_i(t)$, where i goes from 1 to n. A smooth surface immersed in E_3 is a collection of three smooth real-valued function of two variables x^1 and x^2 (see figure 7).

$$\begin{aligned} y_1 &= y_1(x^1, x^2) \\ y_2 &= y_2(x^1, x^2) \\ y_3 &= y_3(x^1, x^2) \end{aligned}$$

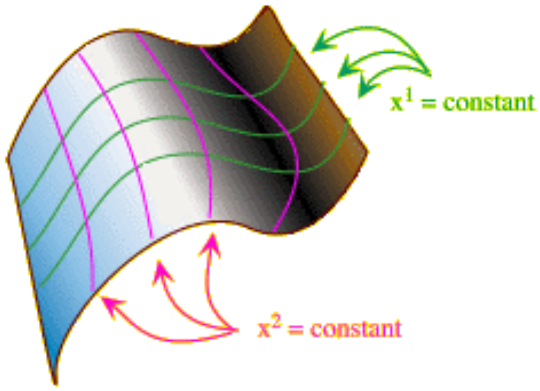

Fig. 7. Smooth surface immersed in E_3

or just

$$y_i = y_i(x^1, x^2), \ \textit{where } i = (1, 2, 3)$$

Then x^1 and x^2 are called the *parameters* or *local coordinates*. A unit sphere ($y_1^2 + y_2^2 + y_3^2 = 1$), when using spherical coordinates, is a good example that illustrates this concept (see figure 8).

$$\begin{aligned} y1 &= \sin(x^1)\cos(x^2) \\ y2 &= \sin(x^1)\sin(x^2) \\ y3 &= \cos(x^1) \end{aligned}$$

where x^1 *and* x^2 are the polar coordinates (the angles are shown on the figure).

Therefore, the parametric equations of a surface show us how to obtain a point on the surface once we know the two local coordinates (or parameters). In other words, this operation allows us to specify a function $E_2 \rightarrow E_3$. Thus, in order to obtain the local coordinates from the Cartesian coordinates y_1, y_2 *and* y_3, one will need to solve for the local coordinates x^i as a function of y_j. For example, for the

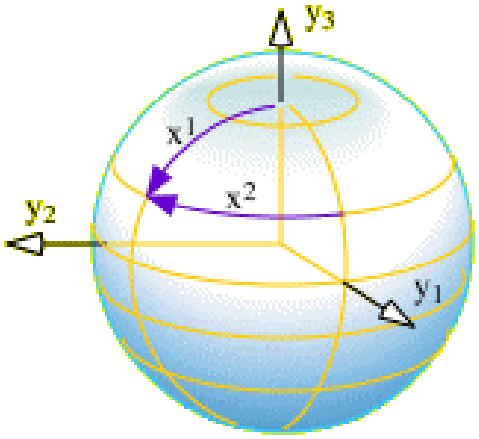

Fig. 8. The unit sphere

sphere case, we will get:

$$x^1 = \cos^{-1}(y_3)$$

$$x^2 = \begin{cases} \cos^{-1}(y_1/\sqrt{y_1^2+y_2^2}) & \text{if } y_2 \geq 0 \\ 2\pi - \cos^{-1}(y_1/\sqrt{y_1^2+y_2^2}) & \text{if } y_2 < 0 \end{cases}$$

This technique presents the advantage of allowing us to give each point on much of the sphere *two unique coordinates*, x^1 and x^2. Even though there is a continuity problem as y_2 approaches 0, since then x^2 "jumps" from 0 to 2π, and at the poles ($y_1 = y_2 = 0$) since the function is not even defined, we still can restrict the portion of the sphere to an open subset of the sphere by having:

$$0 < x^1 < \pi \text{ and } 0 < x^2 < 2\pi$$

x^1 and x^2 are called the *coordinates functions*.

A chart of a surface S is a pair of functions $x = (x^1(y_1, y_2, y_3), x^2(y_1, y_2, y_3))$ which specify each of the local coordinates (parameters) x^1 and x^2 as smooth functions of a general point (global or ambient coordinates) on the surface (see figure 9).

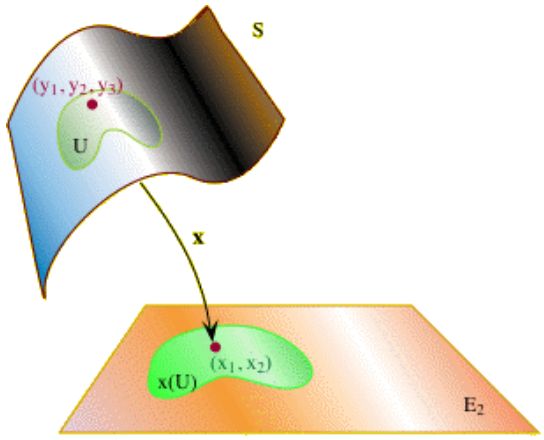

Fig. 9. Chart

2. Smooth Manifolds and Scalar Fields

An *open cover* of $M \subset E_s$ is a collection $\{U_\alpha\}$ of open sets in M such that $M = \cup_\alpha U_\alpha$. The following are two examples that can be used to illustrate these concepts: E_s can be covered by open balls and the unit sphere in E_s can be covered by the collection $\{U_1, U_2\}$ (see figure 10) where:

$$U_1 = \{(y_1, y_2, y_3) | y_3 > -\frac{1}{2}\}$$

$$U_2 = \{(y_1, y_2, y_3) | y_3 < \frac{1}{2}\}$$

A subset M of E_s is called an *n-dimensional smooth manifold* if there is a collection $\{U_\alpha; x^1_\alpha, x^2_\alpha, ..., x^n_\alpha\}$ where:

- The U_α form an open cover of **M**
- Each x^r_α is a C^∞ real-valued function defined on U (that is, $x^r_\alpha : U_\alpha \rightarrow E_n$) given by x(u)=$(x^1_\alpha(u), x^2_\alpha(u), ..., x^n_\alpha(u))$ is one-to-one (that is , to each point U_α is assigned a *unique* set of n coordinates). The tuple $(U_\alpha; x^1_\alpha, x^2_\alpha, ..., x^n_\alpha)$ is called a *local chart of M*. The collection of all charts is called a *smooth atlas of M*.

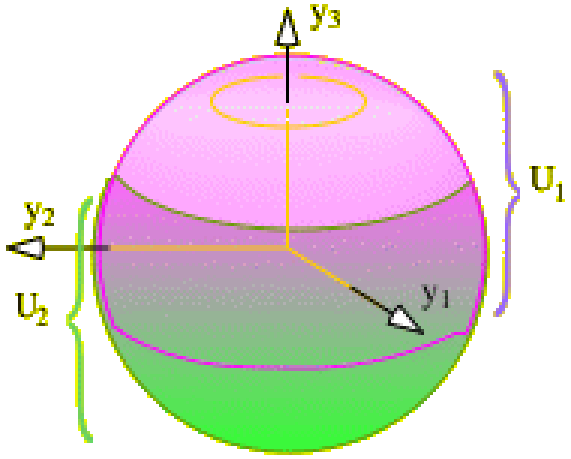

Fig. 10. Unit sphere covered by the collection $\{U_1, U_2\}$

Further, U_α is called a *coordinate neighborhood*.

- If (U, x^j) and $(V, \overline{x}^j)$ are two local charts of M, and if $U \cap V \neq 0$, then one can write $x^i = x^i\overline{x}^j$ and its inverse $\overline{x}^k = \overline{x}^k(x^l)$ for each i and k, where all functions are C^∞ (all the functions are smooth). This ensemble of functions is called the *change-of-coordinates* transformation (see figure 11). One should note that x^i should always be thought as the *local coordinates* (or parameters) of the manifold. It is easy to parameterize each of the open sets U by using the inverse function of x, which assigns to each point in some neighborhood of E_n a corresponding point onto the manifold.

Also it is important to notice that the third condition implies that:

$$det\left(\frac{\delta\overline{x}^i}{\delta x^j}\right) \neq 0$$

$$det\left(\frac{\delta x^i}{\delta\overline{x}^j}\right) \neq 0$$

since the matrices associated to those determinants must be invertible.

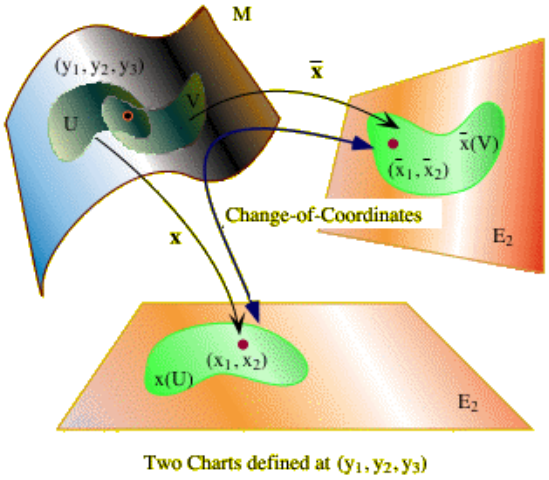

Fig. 11. Change of coordinates using two charts

The following example is a good illustration of the previous definitions. Let's have S^l the unit circle, with the exponential map. Hence, this unit circle is a 1-dimensional manifold.

One has:

$$x : S^1 - \{(1,0)\} \to E_1$$

$$x : S^1 - \{(-1,0)\} \to E_1,$$

with 0¡x, $\overline{x} < 2\pi$, and the change of coordinate maps are given by:

$$\overline{x} = \begin{cases} x + \pi & \text{if } x < \pi \\ x - \pi & \text{if } x > \pi \end{cases}$$

and

$$x = \begin{cases} \overline{x} + \pi & \text{if } \overline{x} < \pi \\ \overline{x} - \pi & \text{if } \overline{x} > \pi \end{cases}$$

Notice the symmetry between x and $\overline{x}$. Also, notice that these change of coordinate functions are only defined when $\theta \neq 0, \pi$.

3. Tangent Vectors and the Tangent Space

In order for us to fully understand the concept of *vector tangent to smooth manifold*, we first need to introduce the concept of *smooth paths* on M.

A smooth path in the smooth manifold M is a smooth map $r : (-a, a) \rightarrow M$, where $r(t) = (y_1(t), y_2(t), ..., y_s(t))$. We say that r is a smooth path through $m \in M$ if $r(t_0) = m$ for some t_0. We can specify a path in M by its coordinates: $y_1 = y_1(t), y_2 = y_2(t), ..., y_s = y_s(t)$.

Since the ambient and local coordinates are functions of each other, we can also express a path in terms of its local coordinates: $x^1 = x^1(t), x^2 = x^2(t), ..., x^n = x^n(t)$.

A *tangent vector* is an operator that maps functions on the manifold.

$$t : \{f : f \in \mathfrak{F}\} \tag{2.15}$$

A good illustration of this is the following example (see figure 12); Let M be the surface $y_3 = y_1^1 + y_2^2$, which can be parameterize by:

$$y_1 = x^1$$

$$y_2 = x^2$$

$$y_3 = (x^1)^2 + (x^2)^2$$

This corresponds to the single chart $(U = M; x^1, x^2)$, where

$$x^1 = y_1 \; and \; x^2 = y_2$$

To specify the tangent vector, let's first specify a path in M, such as:

$$y_1 = \sqrt{t}\sin(t)$$

$$y_2 = \sqrt{t}\cos(t)$$

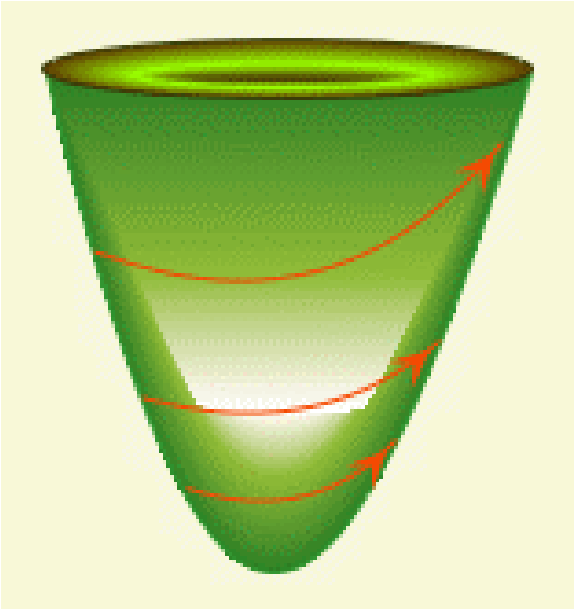

Fig. 12. Tangent vector to a cone. Path on M

$$y_3 = t$$

giving us the path shown in figure 12 and figure 13.

We then can obtain a tangent vector field along the path by taking the appropriate derivatives:

$$(\frac{dy_1}{dt}, \frac{dy_2}{dt}, \frac{dy_3}{dt}) = (\sqrt{t}\cos t + \frac{\sin t}{2\sqrt{t}}, -\sqrt{t}\sin t + \frac{\cos t}{2\sqrt{t}}, 1)$$

In order to get the actual tangent vectors at points in M, one needs to evaluate this at a fixed point t_0. Also, it is easy to realize that we can express the coordinates x^i in terms of t.

If M is an n-dimensional manifold, and $m \in M$, then *the tangent space at* m (see figure 14) is the set T_m of all tangent vectors at m. The above constructions turn T_m into a *vector space*.

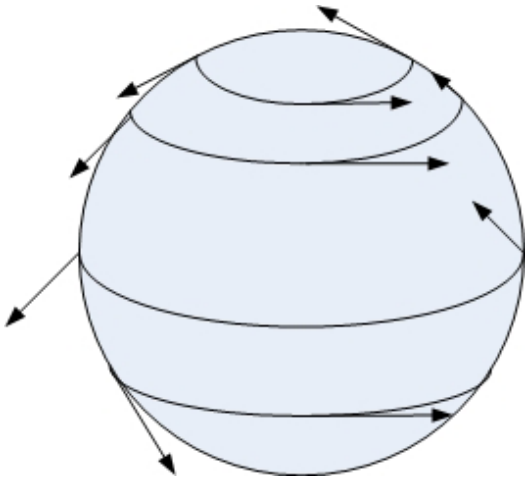

Fig. 13. Tangent vector to a sphere. Path on M

4. Contravariant and Covariant Vector Fields

A *contravariant vector* at $m \in M$ is a collection v^i of n quantities (defined for each chart at m) which transform according to the formula:

$$\overline{v}^i = \frac{\delta \overline{x}^i}{\delta x^j} v^j$$

It goes along with the fact that contravariant vectors are just tangent vectors.

At each point m in a manifold M, we have the n vectors $\frac{\delta}{\delta x^1}, \frac{\delta}{\delta x^2}, ..., \frac{\delta}{\delta x^n}$, where the typical vector $\frac{\delta}{\delta x^i}$ was obtained by taking the derivative of the path. Hence,

$$\frac{\delta}{\delta x^i} = vector\ obtained\ by\ differentiating\ the\ path\ x^j = \begin{cases} t + constant & \text{if j=i} \\ constant & \text{if } j \neq i \end{cases}$$

where the constants are chosen to make $x^i(t0)$ correspond to m for some t_0.

Note that a tangent filed is a field on (part of) a manifold, and as such, it is not, in general, constant. The only things that are constant are its coordinates under the specific chart x. The corresponding coordinates under another chart $\overline{x}$ are $\delta \overline{x}^j / \delta x^i$,

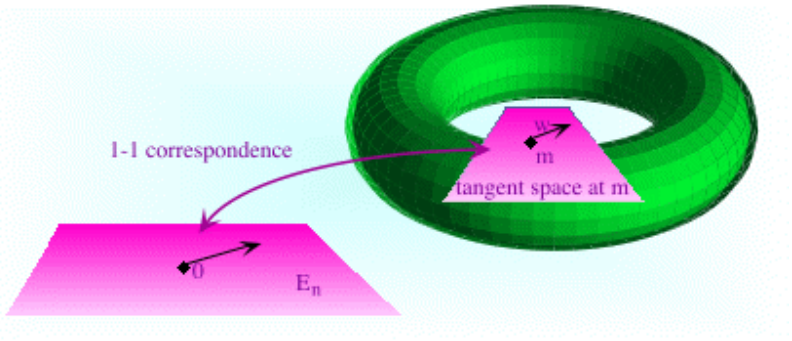

Fig. 14. Tangent space at m

are not constant in general. The vector field looks like figure 15. If one wants to *path together* local vector fields, making them not local anymore, one will need to extend them to the whole of M. In order to do so, one will need to make them zero near the boundary of the coordinate patch. The following procedure will allow this.

If $m \in M$ and x is any chart of M, lat x(m) = y and let D be a ball of some radius r centered at y entirely contained in the image of x. Now define a vector field on the whole of M by:

$$w(p) = \begin{cases} \delta/\delta x^j e^{-R^2} & \text{if p is in D} \\ 0 & \text{otherwise} \end{cases}$$

where R=$(|x(p) - y|)/(r - |x(p) - y|)$. The figure 16 shows what this field looks like on M. The fact that $\overline{V}^i$ is a smooth function of the $\overline{x}^i$ now follows from the fact that all the partial derivatives of all orders vanish as you leave the domain of x.

A *covariant vector field* C on M associates with each chart x a collection of n smooth functions $C_i(x^1, x^2, ..., x^n)$ which satisfy (covariant vector transformation rule):

$$\overline{C}_i = C_j \frac{\delta x^j}{\delta \overline{x}^i}$$

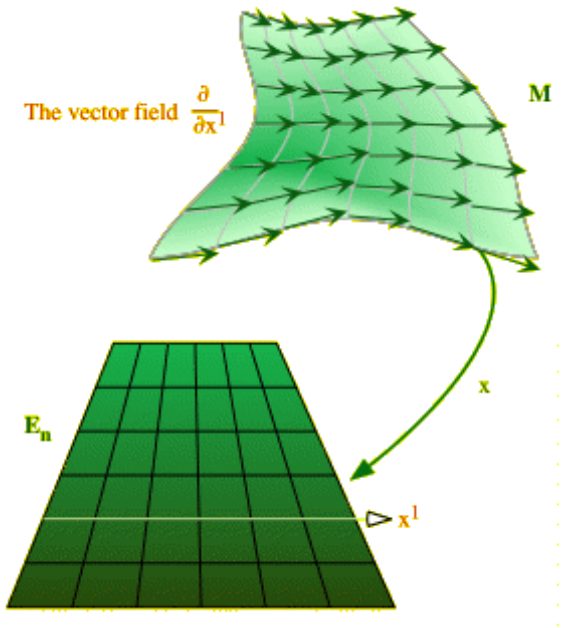

Fig. 15. The vector field $\delta/\delta x^i$

Geometrically, a contravariant vector is a vector that is tangent to the manifold. Hence, a *smooth 1-form*, also called a smooth cotangent vector field on the manifold M (or on an open subset U of M) is a function F that assigns to each tangent vector field V on M (or on the subset U) a scalar field F(V) which is smooth (see figure 17).

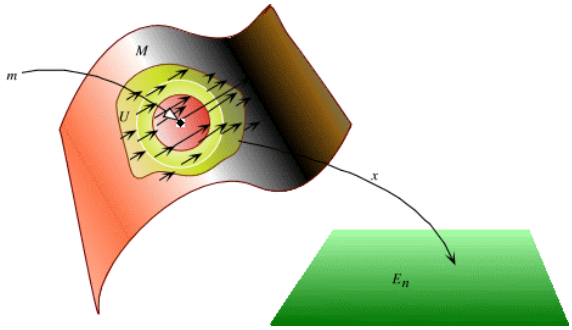

Fig. 16. Field on M

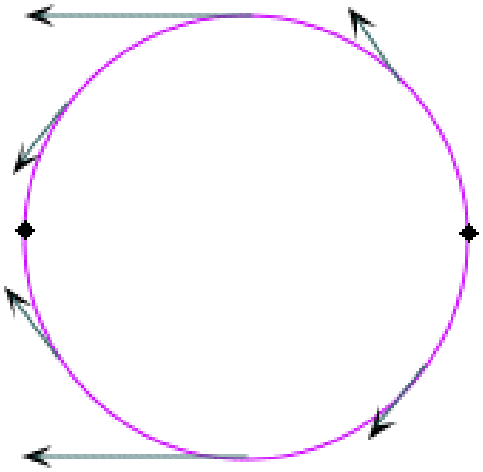

Fig. 17. Smooth cotangent vector field

It has the following properties:

$$F(V+W) = F(V) + F(W)$$

$$F(\alpha V) = \alpha F(V)$$

for every pair of tangent vector fields V and W, and every scalar α. The equivalent in linear algebra is that F is a linear transformation from the vector space of smooth tangent vector fields on M to the the vector space of smooth scalar fields on M.

5. Tensor Fields

Suppose that v = (v1, v2, v3) and w = (w1, w2, w3) are vector fields on E_3. Then their *tensor product* is defined to consist of the nine quantities $v_i w_j$. Thus, let V and W be contravariant, and let C and D be covariant. Then:

$$\overline{v}^i \overline{w}^j = \frac{\delta \overline{x}^i}{\delta x^k} V^k \frac{\delta \overline{x}^j}{\delta x^m} W^m = \frac{\delta \overline{x}^i}{\delta x^k} \frac{\delta \overline{x}^j}{\delta x^m} V^k W^m$$

and similarly,

$$\overline{v}^i \overline{c}_j = \frac{\delta \overline{x}^i}{\delta x^k} \frac{\delta \overline{x}^j}{\delta x^m} V^k C_m$$

and,

$$\overline{c}_i \overline{d}_j = \frac{\delta x_k}{\delta \overline{x}^j} \frac{\delta x_m}{\delta \overline{x}^j} C_k D_m$$

These product fields are called respectively "tensors" of type (2,0), (1,1), and (0,2).

A tensor field of type (2,0) on the n-dimensional smooth manifold M associates with each chart x a collection of n^2 smooth functions $T^{ij}(x^1, x^2, ..., x^n)$ which satisfy the transformation rules shown below. Similarly, we define tensor fields of type (0,2), (1,1), and, more generally, a tensor field of type (m,n).

It is important to note that a tensor field of type (1,0) is just a contravariant vector field, while a tensor field of type (0,1) is a covariant vector field. Similarly, a tensor field of type (0,0) is a scalar field. Also type (1,1) tensors correspond to linear transformations in linear algebra.

The *Kronecker Delta Tensor*, given by:

$$\delta^i_j = \begin{cases} 1 & \text{if j=i} \\ 0 & \text{otherwise} \end{cases}$$

is a tensor field of type (1,1). Indeed, one has $\delta^i_j = \frac{\delta x^i}{\delta x^j}$ and the latter quantities transform according to the rule:

$$\overline{\delta}^i_j = \frac{\delta \overline{x}^i}{\delta x^k} \frac{\delta x^m}{\delta \overline{x}^j} \delta^k_m$$

whence they constitute a tensor field of type (1,1).

A *metric tensor* (see figure 18) is defined by a set of quantities g_{ij} where:

$$g_{ij} = \frac{\delta}{\delta x^i} \frac{\delta}{\delta x^j}$$

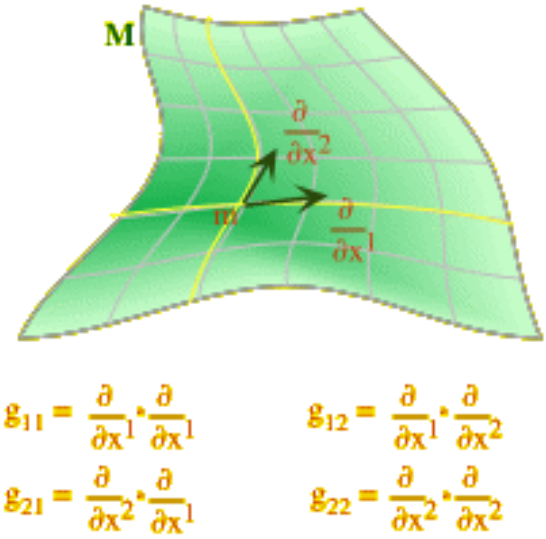

Fig. 18. A metric tensor

This is a tensor of type (0,2). This tensor is called "the metric tensor inherited from the imbedding of M in Es."

6. Riemannian Manifolds

A smooth inner product on a manifold M is a function $< -, - >$ that associates to each pair of smooth contravariant vector fields X and Y a scalar (field) $< X, Y >$, satisfying the following properties.

- Symmetry: $< X, Y >=< Y, X >$ for all X and Y.
  Also, $< aX, bY >$ = ab $< X, Y >$ for all X and Y, and scalars a and b
- Bilinearity: $< X, Y + Z >=< X, Y > + < X, Z >$.
- Non-degeneracy: If $< X, Y >= 0$ for every Y, then X = 0

A manifold endowed with a smooth inner product is called a *Riemannian manifold*. It is important to realize that if x is any chart, and p is any point in the domain of

x, then :

$$< X, Y >= X^i Y^j < \frac{\delta}{\delta x^i}, \frac{\delta}{\delta x^j} >$$

This gives us smooth functions

$$g_{ij} =< \frac{\delta}{\delta x^i}, \frac{\delta}{\delta x^j} >$$

such that

$$< X, Y >= g_{ij} X^i Y^j$$

and which constitutes the coefficients of a type (0,2) symmetric tensor. This tensor is called the *fundamental tensor* or metric tensor of the Riemannian manifold.

Here are some things we can do with a Riemannian manifold. If X is a contravariant vector field on M, then the square norm norm of X is defined by:

$$\|X\|^2 =< X, X >= g_{ij} X^i X^j$$

Unlike in regular algebra, $\|X\|^2$ can be negative. If $\|X\|^2 < 0$, X is called *timelike*; if $\|X\|^2 > 0$, X is called *spacelike*, and if $\|X\|^2 = 0$ X is called *null* (see figure 19). If X is not spacelike, it can then be defined as:

$$\|X\| = (\|X\|^2)^{1/2} = (g_{ij} X^i X^j)^{1/2}$$

One of the useful applications for these Riemannian manifolds is *arc length*. If C is a non-null path in M, then its length is defined as follows: First break the path into segments S where each of them lie in some coordinate neighborhood, and then define the length of S by:

$$L(a, b) = \int_a^b (\pm g_{ij} \frac{dx^i}{dt} \frac{dx^j}{dt})^{1/2}$$

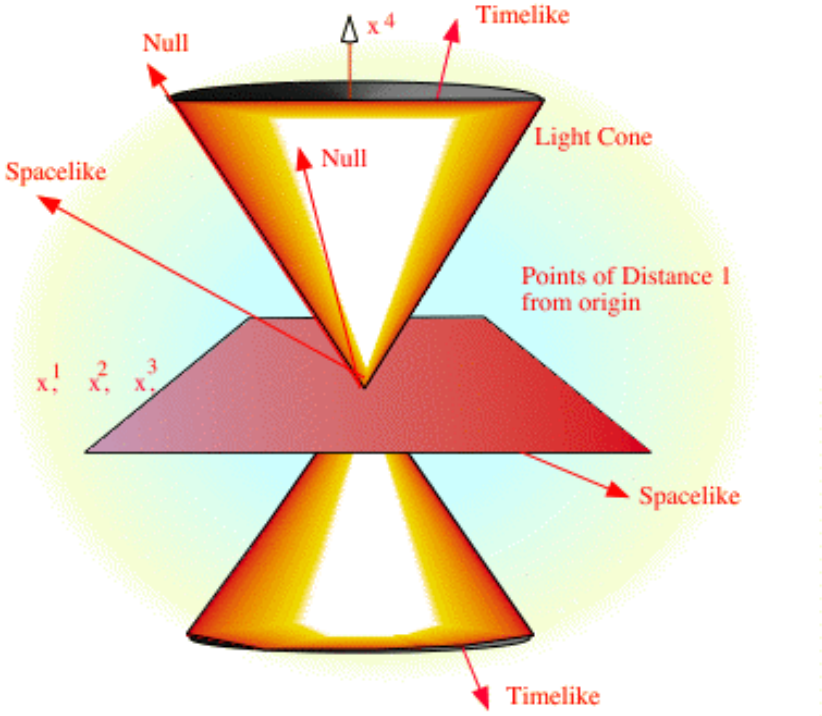

Fig. 19. X as a timelike, spacelike, and null

where the sign ± 1 is chosen as $+1$ if the curve is space-like and -1 if it is time-like. In other words, we are defining the arc-length differential form by:

$$ds^2 = \pm g_{ij} dx^i dx^j$$

Another useful application is the *parameterizations by arc length.* In order to so, one needs to let C be a non-null path $x^i = x^i(t)$ in M. Then, one needs to fix a point t = a on this path, and define a new function s (arc length) by s(t) = L(a, t) = length of path from t = a to t. Then s is an invertible function of t, and, using s as a parameter, $\|dx^i/ds\|^2$ is constant, and equals 1 if C is space-like and -1 if it is time-like.

CHAPTER III

ROBUSTNESS MODELS FOR STATIONARY SYSTEMS: BIASED AND UNBIASED PERTURBATIONS

A robustness quantification and analysis framework, based on differential geometric tools is presented in this chapter. This research shows that when one models imperfectly known quantities as elements of a non-Euclidean manifold, the use of tangent space is an effective method of measuring robustness for a variety of communications and signal processing algorithms. Through the use of different types of parameter surfaces, the latter part of the chapter illustrates the technique.

A. Introduction

Engineers do not always deal with exact knowledge of both design and analysis. Real world applications of algorithms are often made more difficult by the presence of inexact knowledge of certain quantities. Often, such knowledge pertains to the statistics of an underlying random process such as elements of the covariance matrix. For an algorithm to be successful, it must possess a degree of robustness to this inexact knowledge (i.e., the algorithm's performance should not be too sensitive to inexact statistical knowledge). The recognition of the importance of robustness allows engineers to combine both performance and robustness into the development of an algorithm [48, 49, 50, 51, 52]. Much past work has been performed in engineering robustness research by Huber-Strassen [2, 3] on saddle-point techniques. Even though this technique still plays an important role in today's research, it contains a major limitation through the absence of a natural quantitative way to evaluate the performance of the technique. This is a major problem because often the saddle-point solution (defined as the "robust" solution) is difficult to obtain in practical

applications, and thus alternative algorithms were considered.

More recently, a new group of researchers (Halverson *et al*) has started to use differential geometric tools to develop more natural quantitative measures of the degree of robustness. This new direction of research allows engineers to combine both performance and robustness into the development of an algorithm. The thrust of this work is to extend the differential geometric approach toward robustness in different ways.

It is important to note that past work made use of both Euclidean and curved space to model the inexact knowledge. Much of this past work viewed the corresponding perturbations *only* in Euclidean space [8, 9, 12, 13, 14, 16]. For example, if one is considering a strictly increasing nominal distribution function, this distribution function can be allowed to vary locally in a region of m-dimensional Euclidean space, where m is ultimately allowed to approach infinity. There are, however, other situations where the imperfectly known quantities must be constrained to lie on a non-Euclidean manifold.

As an example of practical scenarios where non-Euclidean manifolds may arise, consider applications which involve Gaussian-distributed data. The Gaussian covariance matrix is crucial to an analysis of such an application; but it is unrealistic to assume we will know the matrix perfectly. The set of possible entries for the matrix is imperfectly known but its constraint, such as being positive definite, might be known. This and other possible constraints will result in the entries lying in a manifold which may very well be non-Euclidean.

B. Overview of Approach

Consider a performance function $P : R^m \rightarrow R$. For example, the covariance matrix entries may lie in a subset of R^m and P might be the Mean squares error (MSE) of a linear estimator employed with data processing an imperfectly known variance matrix. Using the techniques described in [8, 9, 10, 11], one can easily associate the robustness of an algorithm with changes in P as one moves away from the nominal point in R^m.

The most convenient way to do this locally is to look at the rate of change of P as one moves in the manifold M spanned by the imperfectly known quantities, where $\widetilde{C} \subset R^m$, where $\widetilde{C}$ is a set containing all the local coordinates. While potentially misleading, the use of slope is convenient and has been found to be a good indicator of the actual performance, in many situations where the more sensitive additional measure provided by curvature [15, 16, 17] is not necessary.

In the next section, we present the estimation problem under consideration and the techniques introduced by [2] to obtain the design of the robust estimator/predictor.

C. Preliminaries and Problem Statement

Linear prediction [53] is an important topic in communications and signal processing with many practical applications like channel estimation, equalization, data detection, image processing etc. In this section we present the problem of linearly predicting the value of a stationary random process and later extend similar analysis to a non-stationary process. This can be done either forward in time or backward in time. Let us begin with the problem of predicting a future value of a stationary random variable from the observation of past values of the process. In particular we consider

forward linear predictor of order p, which forms the prediction of the value $X[n]$ by a weighted linear combination of the past values $X[n-1], X[n-2], \ldots, X[n-p]$.Thus the linearly predicted value of $X[n]$ is

$$\hat{X}[n] = \sum_{i=1}^{p} k_i X[n-i] \tag{3.1}$$

Where k_i represent the p weights in the linear combination. These weights are called predictor coefficients of the forward linear predictor of order p. The forward prediction error is denoted as

$$e_n = \hat{X}[n] - X[n] \tag{3.2}$$

The Mean Square Error is defined as,

$$MSE = E[e[n]^2] \tag{3.3}$$

We choose k_i such that the MSE is minimized. Applying results from Weiner Filtering or Projection theorem, we get

$$\sum_{i=1}^{p} k_i R_x[i-l] = R_x[l], \forall l = 1, 2, \ldots, p \tag{3.4}$$

$$\begin{pmatrix} R_{xx}[0] & \ldots & R_{xx}[-p+1] \\ \vdots & \ddots & \vdots \\ R_{xx}[p-1] & \ldots & R_{xx}[0] \end{pmatrix} \cdot \begin{pmatrix} k_1 \\ \vdots \\ k_p \end{pmatrix} = \begin{pmatrix} R_{xx}[1] \\ \vdots \\ R_{xx}[p] \end{pmatrix}$$

These are called the Yule-Walker [54, 55] equations, which can be solved to obtain k_i. Thus the design of the Linear Predictor depends on the underlying joint distribution, and specifically the correlation matrix. Often the correlation matrix is imperfectly known and thus the choice of k_i arises from the nominal values of the entries in the correlation matrix. Once the k_i are determined to give the maximum performance at the nominal, the linear predictor is fixed. However the parameters that determine

the system to be at the nominal point may vary, thus changing the performance of the estimator. Robustness [1, 2, 3] involves the insensitivity of the performance of a system in the presence of a change in the point of operation from the nominal. Here again the definition of performance is not fixed. Performance can be computed from the MSE or Absolute Error or any other metric.

In the current research we use MSE as the performance metric. Consider a Linear Predictor of order 2,

$$\hat{X}[3] = k_1 X[1] + k_2 X[2] \tag{3.5}$$

For a Wide Sense Stationary Process, the nominal correlation matrix is

$$C\Big|_{WSS} = \begin{pmatrix} a & b/\sqrt{2} & c/\sqrt{2} \\ b/\sqrt{2} & a & b/\sqrt{2} \\ c/\sqrt{2} & b/\sqrt{2} & a \end{pmatrix}$$

Applying Projection Theorem or Yule-Walker Equations, we obtain k_1 and k_2. The MSE is defined as

$$\begin{aligned} MSE &= E[[\hat{X}[3] - X[3]]^2] \\ &= E[[k_1 X[1] + k_2 X[2] - X[3]]^2] \\ &= a[1 + k_1^2 + k_2^2] + \sqrt{2} b k_2 [k_1 - 1] - \sqrt{2} c k_1 \\ &= P(a, b, c) \end{aligned} \tag{3.6}$$

Thus the MSE depends on the correlations of the random variables $X[3]$, $X[2]$ and $X[1]$. The parameters that determine performance are therefore the entries of the correlation matrix of the system. The predictor defined above can be viewed as a parametric system since for designing an optimal predictor; we need to know the exact correlation matrix values of the random variables $X[1]$, $X[2]$ and $X[3]$. This is

however a highly optimistic design and in most cases it is difficult to know the exact values. Thus we have to decide on a nominal value of the correlation matrix that can be used. The actual values of the elements of the matrix may be different from the nominal and hence the predictor may no longer remain optimal. We need to know how much the system performance changes as the values of (a, b, c) change. Here we define robustness as the sensitivity of the system performance to the change in design parameters (covariance matrix in this case). Loosely, the system (linear predictor) is robust if the performance does not change much with the values of the covariance matrix elements.

Since the MSE directly gives us the performance we use MSE equation and obtain the performance function $P(a, b, c)$. The MSE is defined as

$$MSE = a[1 + k_1^2 + k_2^2] + \sqrt{2}bk_2[k_1 - 1] - \sqrt{2}ck_1$$

In many situations, it is possible that we have knowledge of some functional relation between the elements of the covariance matrix instead of simple affine space. We would like to make use of this extra information in the analysis of the system performance and robustness.

$$c = f(a, b) \tag{3.7}$$

This parameter surface thus formed is shown in figure 20. The information given by the parameter equation can be used along with the performance function and we can reduce P to a function of only two variables.

We have a 2-dimensional surface embedded in 3-dimensional space. This performance surface thus formed is shown in figure 21. The height of the surface above the $a-b$ plane at any point gives the MSE for that point. Note that here we are writing c as a function of a and b explicitly, and our results depend on this assignment of

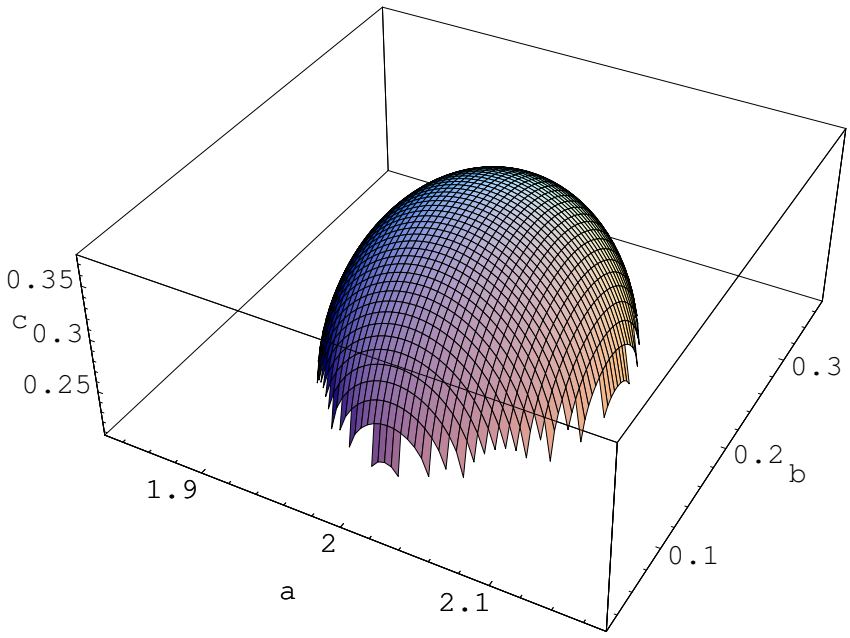

Fig. 20. Parameter surface example (upper)

independent and dependent variables. The choice seems reasonable since we are more likely to know something about a and b because they involve autocorrelations (a is $R_{xx}[0]$) and closer correlation (b is $R_{xx}[1]$). The less understood c (since is a farther correlation, $R_{xx}[2]$) is then allowed to depend on a and b. But this assignment is not unique. With this approach we are allowing free variation of a and b and forcing c to vary according to its constraint. This may be a good model in robustness where more is known about a and b. We call this procedure biased perturbation. In the above relation, the values k_1 and k_2 are fixed at design time based on the nominal values of the parameters a, b and c and then we plot the performance surface as the parameters deflect away from the nominal. This means that if the performance surface is almost flat, then the MSE does not vary much with a and b, and thus the system is robust. On the other hand if we have a performance surface that looks like an normal or inverted mountain, then the points near the peak are highly sensitive

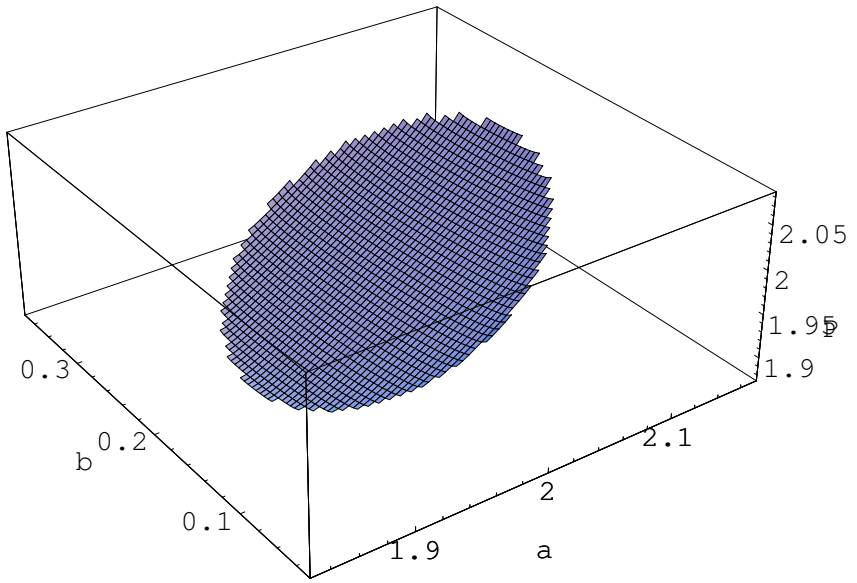

Fig. 21. Performance surface example (upper)

to the parameters a and b, and a small change in the parameters will cause a drastic change in performance, making the system highly un-robust. The simplest way to check the performance change at any point is to measure the slope of the tangent to the performance surface at that point. In the past similar system modeling has been done and slope has been the primary tool used to quantify system robustness. Differentiable manifolds have been shown to be useful in analyzing robustness in many contexts.

As an example of the application of the above approach, consider estimation in zero mean data with two samples. The samples are realizations of the random variables X[1] and X[2], where X[1], X[2] and X[3] are jointly Gaussian distributed random variables with covariance matrix C:

$$C = \begin{pmatrix} a & b/\sqrt{2} & c/\sqrt{2} \\ b/\sqrt{2} & a & b/\sqrt{2} \\ c/\sqrt{2} & b/\sqrt{2} & a \end{pmatrix}$$

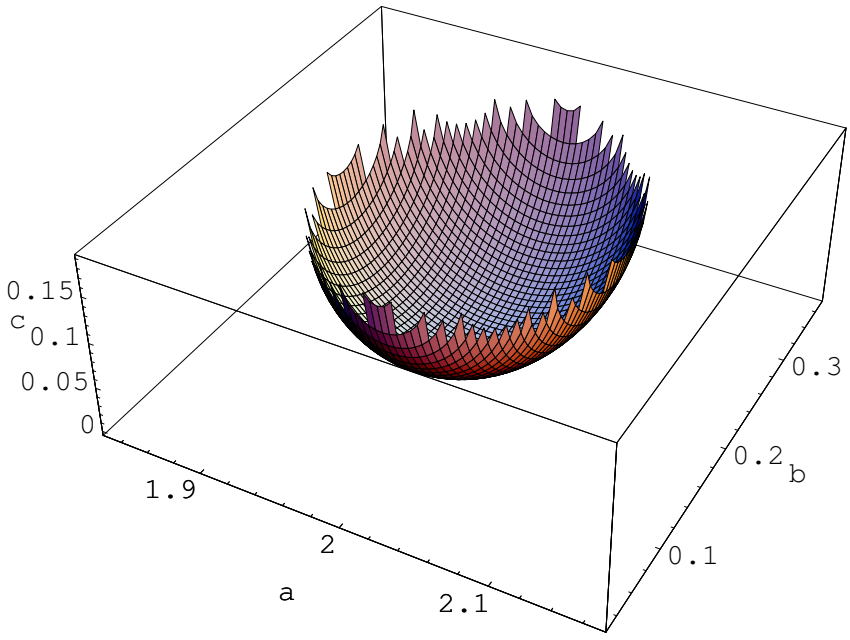

Fig. 22. Parameter surface example (lower)

where $a = R_{xx}[0]$, $b/\sqrt{2} = R_{xx}[1]$ and $c/\sqrt{2} = R_{xx}[2]$. We need to ensure a positive definite matrix. Performance function is the MSE.

Supposed (a, b, c) are close to the nominal (a_0, b_0, c_0) according to (for reasons discussed later):

$$3(a - a_0)^2 + 2(b - b_0)^2 + (c - c_0)^2 = {\epsilon_S}^2 \tag{3.8}$$

where equation 3.8 represents a surface parameter of a 3D manifold centered at (a_0, b_0, c_0). The manifold of vectors of covariance entries is a paraboloidal bowl (an ellipsoidal shape) which is constrained via positive definite covariance matrix. Combining the constraint of the previous equation together with the need of a positive definite covariance matrix results in (a, b, c) lying in a two dimensional manifold.

The following section is dedicated to the derivations of closed form expressions for the measure that we are going to use for the robustness analysis.

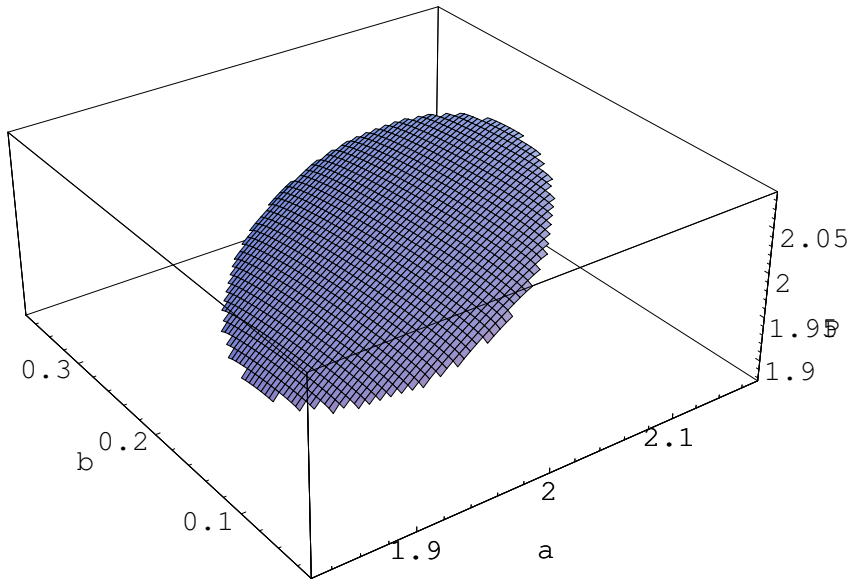

Fig. 23. Performance surface example (lower)

D. Derivation of Formulas Used for Robustness Analysis

In this section, we first define and then illustrate the derivation of Varma and Halverson [18] for expressions for the slopes that we are going to use for the robustness analysis, namely slope with biased perturbations and slope with unbiased perturbations. We remark that the way a perturbation about the nominal in a manifold is modeled is not unique, and [18] describes two important ways (biased and unbiased) that thus may occur.

1. General Worst Case Gradient to Riemannian Manifold

Consider the manifold M and consider a point P on M, with local coordinates $(u_i)_i$. Consider the Riemannian metric g on the tangent space T_pM. If $\overrightarrow{X}$ and $\overrightarrow{Y}$ are tangent vectors at P, then:

$$\overrightarrow{X} = \sum_i X_i \frac{\partial}{\partial u_i} \tag{3.9}$$

$$\overrightarrow{Y} = \sum_i Y_i \frac{\partial}{\partial u_i} \tag{3.10}$$

therefore:

$$\langle \overrightarrow{X}, \overrightarrow{Y} \rangle = \sum_i g_{ij} X_i Y_j \tag{3.11}$$

where:

$$g_{ij} = g\Big(\frac{\partial}{\partial u_i}, \frac{\partial}{\partial u_j}\Big) \tag{3.12}$$

Thus, g is a covariant tensor of order 2. For tangent vector, $\overrightarrow{X} = \sum_i X_i \frac{\partial}{\partial u_i}$, by varying $\overrightarrow{X}$ we wish to maximize directional derivative of function h at P,

$$D_{\overrightarrow{X}} h(P) = \sum_i X_i \frac{\partial h}{\partial u_i}\Big|_P \tag{3.13}$$

subject to the condition that $\overrightarrow{X}$ have a unit length.

$$\langle \overrightarrow{X}, \overrightarrow{Y} \rangle = 1 \tag{3.14}$$

and

$$\sum_i g_{ij} X_i Y_j = 1 \tag{3.15}$$

Using the Lagrange multiplier method, we get:

$$J = \sum_i X_i \frac{\partial h}{\partial u_i}\Big|_P - \lambda \sum_{i,k} g_{jk} X_j X_k \tag{3.16}$$

thus:

$$\forall i,\ \frac{\partial J}{\partial X_i} = \frac{\partial}{\partial X_i}\Big(\sum_i X_i \frac{\partial h}{\partial u_i}\Big|_P - \lambda(\sum_{j,k} g_{jk} X_j X_k)\Big)$$

then

$$\frac{\partial}{\partial X_i}\Big(\sum_i X_i \frac{\partial h}{\partial u_i}\Big|_P - \lambda(\sum_{j \neq k} g_{jk} X_j X_k) - \lambda(\sum_j g_{jj} X_j X_j)\Big) = 0$$

hence

$$\frac{\partial h}{\partial u_i}\Big|_P - \lambda \sum_{k \neq i} g_{ik} X_k - \lambda \sum_{j \neq i} g_{ij} X_j - 2\lambda g_{ii} X_i = 0 \tag{3.17}$$

Using the fact that $G = (g_{ij})$ is symmetric, we have then:

$$\frac{\partial h}{\partial u_i}\Big|_P - 2\lambda \sum_{j \neq k} g_{ij} X_j - 2\lambda g_{ii} X_i = 0 \tag{3.18}$$

or:

$$\frac{\partial h}{\partial u_i}\Big|_P - 2\lambda \sum_{j} g_{ij} X_j = 0 \tag{3.19}$$

$$D_{\overrightarrow{X}} h(P) - 2\lambda.1 = 0 \tag{3.20}$$

$$\lambda = \frac{1}{2} D_{\overrightarrow{X}} h(P) \tag{3.21}$$

$$\frac{\partial h}{\partial u_i}\Big|_P = D_{\overrightarrow{X}} h(P). \sum_{j} g_{ij} X_j \tag{3.22}$$

$$\begin{pmatrix} \partial h/\partial u_1 \\ \vdots \\ \partial h/\partial u_m \end{pmatrix} = D_{\overrightarrow{X}} h G \overrightarrow{X}$$

where G is a $m * m$ matrix of the following form:

$$G = \begin{pmatrix} g_{11} & g_{12} & \dots \\ g_{21} & g_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

and where $\overrightarrow{X}$ is a column vector of the following form:

$$X_1 = \begin{pmatrix} X_1 \\ \vdots \\ X_m \end{pmatrix}$$

Therefore:

$$\begin{pmatrix} \partial h/\partial u_1 \\ \vdots \\ \partial h/\partial u_m \end{pmatrix} = D_{\overrightarrow{X}} h.G \begin{pmatrix} X_1 \\ \vdots \\ X_m \end{pmatrix}$$

Since $D_{\overrightarrow{X}}h$ is a scalar, we will obtain:

$$D_{\overrightarrow{X}}h\Big(\frac{\partial h}{\partial u_1}\dots\frac{\partial h}{\partial u_m}\Big)\begin{pmatrix} X_1 \\ \vdots \\ X_m \end{pmatrix} = \Big(\frac{\partial h}{\partial u_1}\dots\frac{\partial h}{\partial u_m}\Big)G^{-1}\begin{pmatrix} \frac{\partial h}{\partial u_1} \\ \vdots \\ \frac{\partial h}{\partial u_m} \end{pmatrix} \tag{3.23}$$

$$(D_{\overrightarrow{X}}h)^2 = \nabla h G^{-1}\nabla h^T \tag{3.24}$$

Hence:

$$(D_{\overrightarrow{X}}h)\Big|_{Extreme} = \sqrt{\nabla h G^{-1}\nabla h^T} \tag{3.25}$$

While it might not be obvious from the above expression, the value of $(D_{\overrightarrow{X}}h)\Big|_{Extreme}$ does not depend on the choice of the coordinates, provided the underlying manifold remains fixed. However, this follows from the classical interpretation of directional derivative as a limit of ∇h over the arc length, where the answer computes to be independent of the curve chosen as long as the tangent to the curve at P is fixed.

2. Gradient with Unbiased Perturbations-Optimal Model

If the parameter surface is defined by $c = f(a, b)$, we can use $a = u_1$ and $b = u_2$. Then we obtain:

$$\frac{\partial}{\partial a} = (1, 0, \frac{\partial c}{\partial a}) \tag{3.26}$$

$$\frac{\partial}{\partial b} = (0, 1, \frac{\partial c}{\partial b}) \tag{3.27}$$

Inheriting the inner product from R^3

$$g_{11} = 1 + \Big(\frac{\partial c}{\partial a}\Big)^2 \tag{3.28}$$

$$g_{22} = 1 + \Big(\frac{\partial c}{\partial b}\Big)^2 \tag{3.29}$$

$$g_{12} = g_{21} = \frac{\partial c}{\partial a}.\frac{\partial c}{\partial b} \tag{3.30}$$

We then remark that this choice of coordinates is for convenience, and the same result will be obtained for an alternative choice. Thus, the result is without bias. If the performance function is $P = h(a, b, c) \equiv h(a, b)$, we then obtain:

$$\left(D_{\overrightarrow{X}}h\right)\Big|_{Extreme} = \sqrt{\nabla h G^{-1} \nabla h^T} \tag{3.31}$$

$$\nabla h = \left(\frac{\partial h}{\partial a} \quad \frac{\partial h}{\partial b}\right) \tag{3.32}$$

and finally:

$$G = \begin{pmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{pmatrix}$$

3. Gradient with Biased Perturbations-Suboptimal Model

In the case where the slope is biased with perturbations, the variations are translated to the Euclidean space. It is with these variations that the bias enters the system, because the manifold has been altered. Thus, $g_{ij} = \delta_{ij}$, where δ_{ij} is the *Kronecker delta*.

This time we have:

$$G = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

G becomes the identity matrix. Using the new matrix G into equation 3.31, we obtain:

$$\left(D_{\overrightarrow{X}}h\right)\Big|_{Extreme} = \sqrt{\nabla h G^{-1} \nabla h^T} = \sqrt{(\frac{\partial h}{\partial a})^2 + (\frac{\partial h}{\partial b})^2} \tag{3.33}$$

The equation (3.33) is the well known version of the directional derivative applying on functions on affine space.

E. Calculations

All the results are obtained in this section with the use of Mathematica simulations.

$$h(a,b,c) \equiv MSE = a[1 + k_1^2 + k_2^2] + \sqrt{2}bk_2[k_1 - 1] - \sqrt{2}ck_1 \tag{3.34}$$

$$c(a,b) = c_0 \pm \sqrt{(\epsilon_1^2 - 3(a-a_0)^2 - 2(b-b_0)^2)} \tag{3.35}$$

1. Detail of Derivations

$$Slope\Big|_{Biased} = \Big(D_{\vec{X}} h\Big)\Big|_{Extreme} = \sqrt{(\frac{\partial h}{\partial a})^2 + (\frac{\partial h}{\partial b})^2} \tag{3.36}$$

$$Slope\Big|_{Biased} = \sqrt{(\frac{\partial P}{\partial a})^2 + (\frac{\partial P}{\partial b})^2} \tag{3.37}$$

$$\frac{\partial}{\partial a} = (1, 0, \frac{\partial c}{\partial a}) \tag{3.38}$$

$$\frac{\partial}{\partial b} = (0, 1, \frac{\partial c}{\partial b}) \tag{3.39}$$

$$g_{11} = 1 + \Big(\frac{\partial c}{\partial a}\Big)^2 \tag{3.40}$$

$$g_{22} = 1 + \Big(\frac{\partial c}{\partial b}\Big)^2 \tag{3.41}$$

$$g_{12} = g_{21} = \frac{\partial c}{\partial a}.\frac{\partial c}{\partial b} \tag{3.42}$$

$$\nabla P = \Big(\frac{\partial P}{\partial a} \quad \frac{\partial P}{\partial b}\Big) \tag{3.43}$$

$$
\begin{aligned}
Slope\Big|_{Un-biased} &= \sqrt{\nabla P G^{-1} \nabla P^T} \\
&= \sqrt{\left(\frac{\partial P}{\partial a} \quad \frac{\partial P}{\partial b}\right) G^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \frac{\partial P}{\partial b} \end{pmatrix}} \\
&= \sqrt{\begin{pmatrix} \frac{\partial P}{\partial a} \\ \frac{\partial P}{\partial b} \end{pmatrix}^T G^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \frac{\partial P}{\partial b} \end{pmatrix}} \\
&= \sqrt{\left(\frac{\partial P}{\partial a} \quad \frac{\partial P}{\partial b}\right) \begin{pmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \frac{\partial P}{\partial b} \end{pmatrix}} \\
&= \sqrt{\frac{\left(\frac{\partial P}{\partial a} \quad \frac{\partial P}{\partial b}\right) \begin{pmatrix} g_{22} & -g_{12} \\ g_{21} & g_{11} \end{pmatrix} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \frac{\partial P}{\partial b} \end{pmatrix}}{g_{11}g_{22} - g_{12}g_{21}}}
\end{aligned} \tag{3.44}
$$

2. Gradient Distributions and Equalizing Factors

a. Gradient Distributions

The gradient over a region of the manifold is not constant. It very much varies over the whole region. Therefore instead of merely computing the average of the values over the region, we seek a method that quickly illustrates the nature of the variations. If we regard gradient as a random variable, then vital information would be carried by its distribution/density and the form factor of the curve. The following procedure is used throughout in the analysis:

- We first decide on the nominal covariance matrix for (X_3, X_2, X_1). We also need to assume additional information about the parameters (a, b, c) which compose the covariance matrix expressed by the parameter surface.

- The actual parameters are unlikely to vary from the nominal by a large amount.

After choosing an area local to a nominal, we vary the parameters around this region, located on the parameter surface. Then, the gradient values for both, the unbiased and the biased, perturbations are calculated for the new system state (point on the performance surface).

- A sample density for these gradient values is obtained.

b. Equalizing Factor

Consider first the sphere in 3-D. If n points are equally arranged along circles of constant latitude, they gradually become more close together as one approaches the pole (figure 24). For this reason when computing a sample density for gradient, such points are over represented as we approach the pole. When counting them their number should be given reduced "factor" [56]. The same issue arises with an ellipsoid. The same "increase in density" of points at poles occurs.

Now, let's consider an ellipsoid ($Ax^2 + By^2 + z^2 = C$), where one views the "poles" generated by rotating an ellipse ($Ax^2 + By^2 = C$) about the long axis (see figure 25). The same "increase in density" of points at poles occurs, but compared to the sphere, the equalizing is more complicated.

The equalizing factor could be computed using the area element [57, 58, 59]. Let's consider the ellipsoid, $Aa^2 + Bb^2 + c^2 = \epsilon^2$ or equivalently $c = \pm\sqrt{(\epsilon^2 - Aa^2 - Bb^2)}$. The Area element is then equal to:

$$Area\ element\ = \sqrt{1 + (\frac{\partial c}{\partial a})^2 + (\frac{\partial c}{\partial b})^2}dadb \tag{3.45}$$

where $\sqrt{1 + (\frac{\partial c}{\partial a})^2 + (\frac{\partial c}{\partial b})^2}$ represents the weight. We then obtain:

$$(Equalizer)^2 = 1 + (\frac{\partial c}{\partial a})^2 + (\frac{\partial c}{\partial b})^2 \tag{3.46}$$

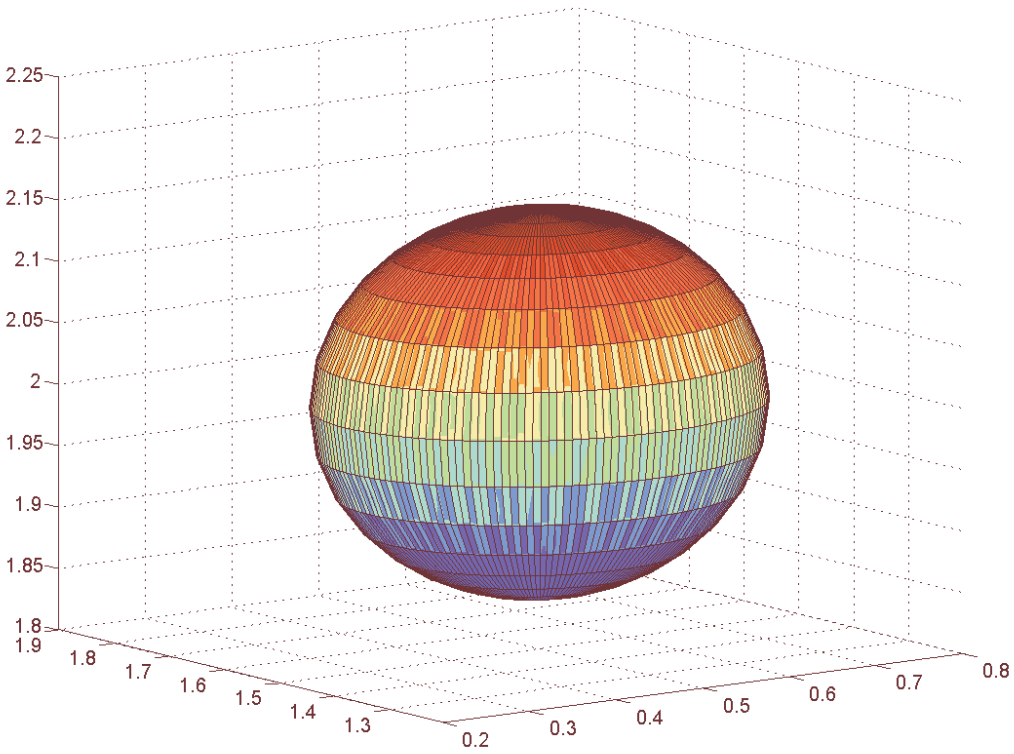

Fig. 24. Ellipsoid

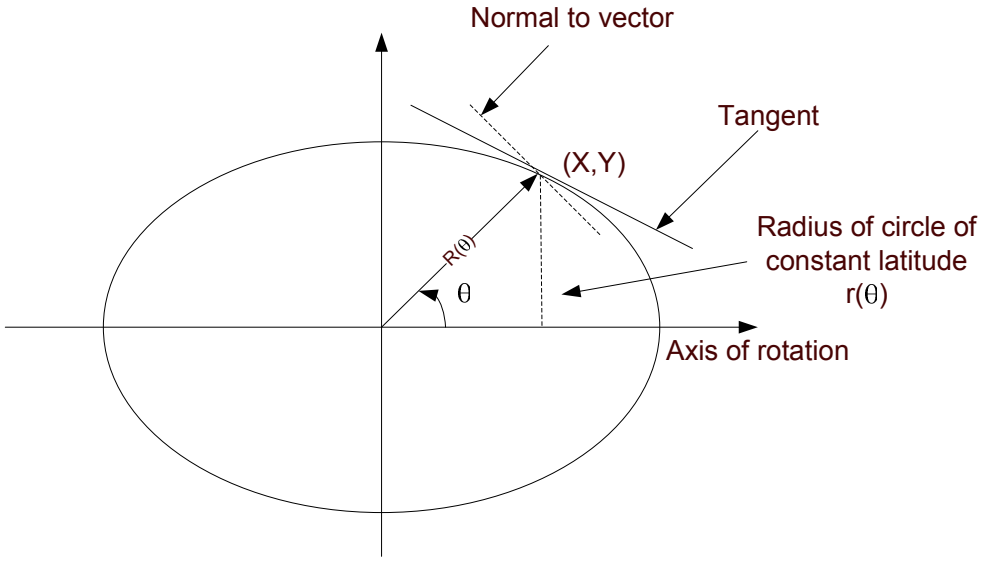

Fig. 25. Rotation along the axis for an ellipsoid

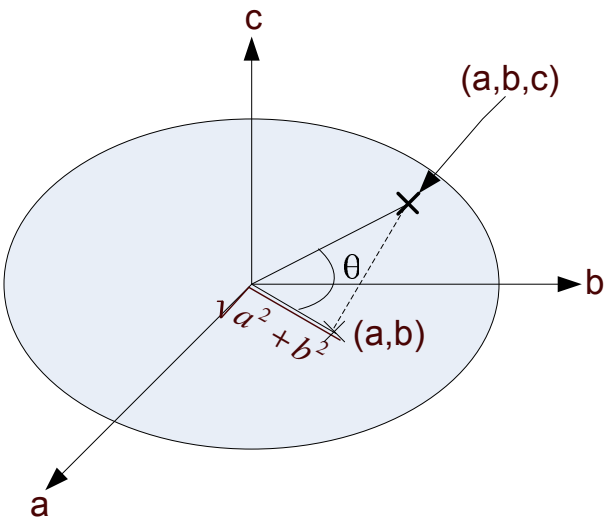

Fig. 26. (a,b,c) related to the latitude θ

Now, replacing c by its corresponding expression, and taking the partial of c with respect to a and b, one can obtain:

$$Equalizer = \sqrt{1 + (\frac{\partial c}{\partial a})^2 + (\frac{\partial c}{\partial b})^2} \tag{3.47}$$

c. Simulation Size and Bin-Size

The Gradient densities are all based on bin-sizes chosen to generate 100 bins. The sample densities are obtained by adding, for each bin, the equalizing factors corresponding to each of the tangent slopes.

The simulation size includes 1000 sets of data points. For each set of three points ($c = f(a, b)$) the corresponding tangent slope to the surface is computed. The following procedure is used all along in the analysis:

- We first decide on the nominal covariance matrix for (X_1, X_2, X_3). We also need

to assume additional information about the parameters (a, b, c) which compose the covariance matrix expressed by the parameter surface.

- The actual parameters are unlikely to vary from the nominal by a large amount. After choosing an area local to a nominal, we vary the parameters around this region, located on the parameter surface. Then, the gradient values for both the unbiased and the biased, perturbations are calculated for the new system state (point on the performance surface).

- Different values of nominal parameters are tested. Also, the effect of the bin-size over the sample densities is implemented. The last case scenario treated is when the variation from the nominal is bigger than usual.

We treated two different examples. The first example corresponds to the use where the covariance matrix lies on a ellipsoidal ball centered on the nominal. Such a parameter surface is bounded with positive Gaussian curvatures. The second example uses a new parameter surface; the correlation $R_{xx}[2]$ is scaled to geometric mean of correlation's $R_{xx}[0]$ and $R_{xx}[1]$ thus yielding a open surface with negative curvature.

F. Computation of Gradient

For example 1 we employ coordinates (a, b) and express the tangent vectors $(\partial/\partial a, \partial/\partial b)$ in terms of the Euclidean vectors. Inherently the inner product from $\mathcal{R}^3$, it is routine to obtain $g_{11} = 1 + (\partial c/\partial a)^2$, $g_{22} = 1 + (\partial c/\partial b)^2$ and $g_{12} = g_{21} = \partial c/\partial a * \partial c/\partial b$ for the unbiased case. For example 2, the same method is obtained, except different parameter surface is employed.

G. Results and Examples

1. Parameter Surface Example 1

In this first example, the parameters (a,b,c) are related by:

$$\left\| \begin{pmatrix} a & b/\sqrt{2} & c/\sqrt{2} \\ b/\sqrt{2} & a & b/\sqrt{2} \\ c/\sqrt{2} & b/\sqrt{2} & a \end{pmatrix} - \begin{pmatrix} a_0 & b_0/\sqrt{2} & c_0/\sqrt{2} \\ b_0/\sqrt{2} & a_0 & b_0/\sqrt{2} \\ c_0/\sqrt{2} & b_0/\sqrt{2} & a_0 \end{pmatrix} \right\| = \|E\| = \epsilon \qquad (3.48)$$

For this example ($3 \star 3$ matrix), let $\|E\|$ represent the value of the natural norm of the generic linear group G1(3,IR) applied to E, also known as the *Frobenius Norm* [60]. Then:

$$\|E\|^2 = 3(a - a_0)^2 + 2(b - b_0)^2 + (c - c_0)^2 \qquad (3.49)$$

representing an ellipsoidal shape. We then impose the constraint that $\|E\|^2 = \epsilon^2$. Various ϵ can be considered, but we always employ an ϵ small enough so that all parameters (a, b, c) correspond to a positive definite matrix. Each corresponding C matrix formed with those points has the particularity of being real and symmetrical forming an Hermitian matrix. The Hermitian matrix has special properties that could be used by one to verify that the sets of points are positive definite. One would only need to compute all the eigenvalues, knowing that if all the eigenvalues are positive, then the matrix is positive definite. Note that any not-positive definite sets of points are automatically dropped by the simulation.

We remark that the proper interpretation of the parameter surface is that the surfaces induces the "nominal" (a_0, b_0, c_0) and not the reverse. The nominal is simply a convenient center of gravity serving as a kind of average value, for choosing the estimator threshold, for example.

Let's have ϵ be the deviation of the covariance matrix from its nominal. Note that

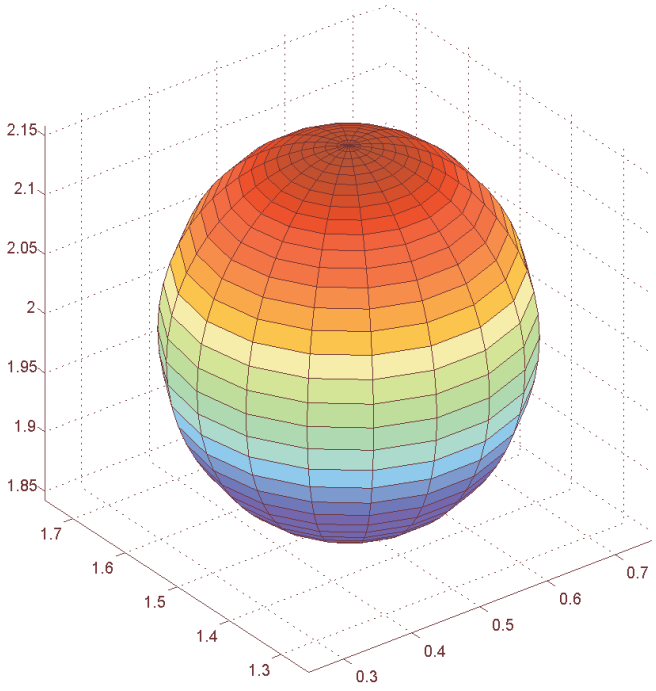

Fig. 27. Picking the right set of points

ϵ is also directly used to verify if the covariance matrix is positive definite (through the computation of each set of coordinates a, b, c).

2. Simulation Results

Distributions of the gradient (probability density) over the manifold are used to visualize if the estimation algorithm is or is not robust. The more "compact" the density curve, the more it shows the absence of big slopes, and the less variable is the gradient. If in addition the point of "compactification" is of small value, then the estimator might be called "robust."

Some of the scenarios that one could potentially encounter are represented in table I. There are eight sets of two graphs that cover all those cases (see figures 28 to 35). For all the densities the bin-size is $4 * 10^{-2}$, over range of 0-4 this generates

Table I. All the different case scenarios for the 1st surface parameter

Candidate	a_0	b_0	c_0	k_1	k_2	ϵ
1	2	0.2	0.2	0.0660409	0.0660409	0.174069
2	2	0.2	-0.2	-0.0760911	0.0760911	0.174069
3	2	-0.2	0.2	0.0660409	-0.0660409	0.174069
4	2	0.5	0.2	0.0407336	0.169576	0.177059
5	2	0.2	0.5	0.17557	0.17264	0.175570
6	4	0.2	0.2	0.034148	0.034148	0.346843
7	2	0.2	0.7	0.243706	0.0534781	0.177271
8	2	0.2	0.2	0.243706	0.0534781	0.348138

100 bins. Each of the sample distributions (see table I) are based on 1000 points.

Table II gives the corresponding means for both the biased and unbiased case, for each case scenario. The weighted mean is used to combine average values from samples of the same population with different sample sizes:

$$\bar{x} = \frac{\sum_{i=1}^{n} w_i.x_i}{\sum_{i=1}^{n} w_i} \tag{3.50}$$

The weights w_i represent the bounds of the partial sample. In other applications they represent a measure for the reliability of the influence upon the mean by respective values.

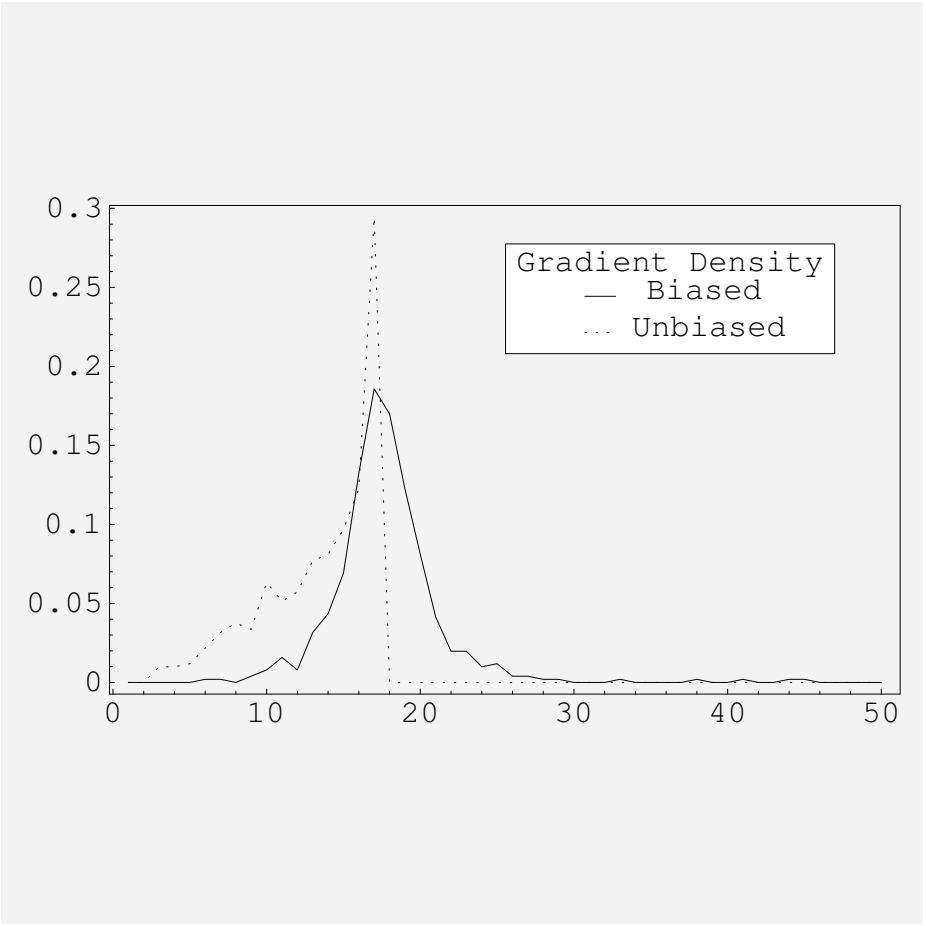

Fig. 28. Gradient densities for the first parameter surface: case 1

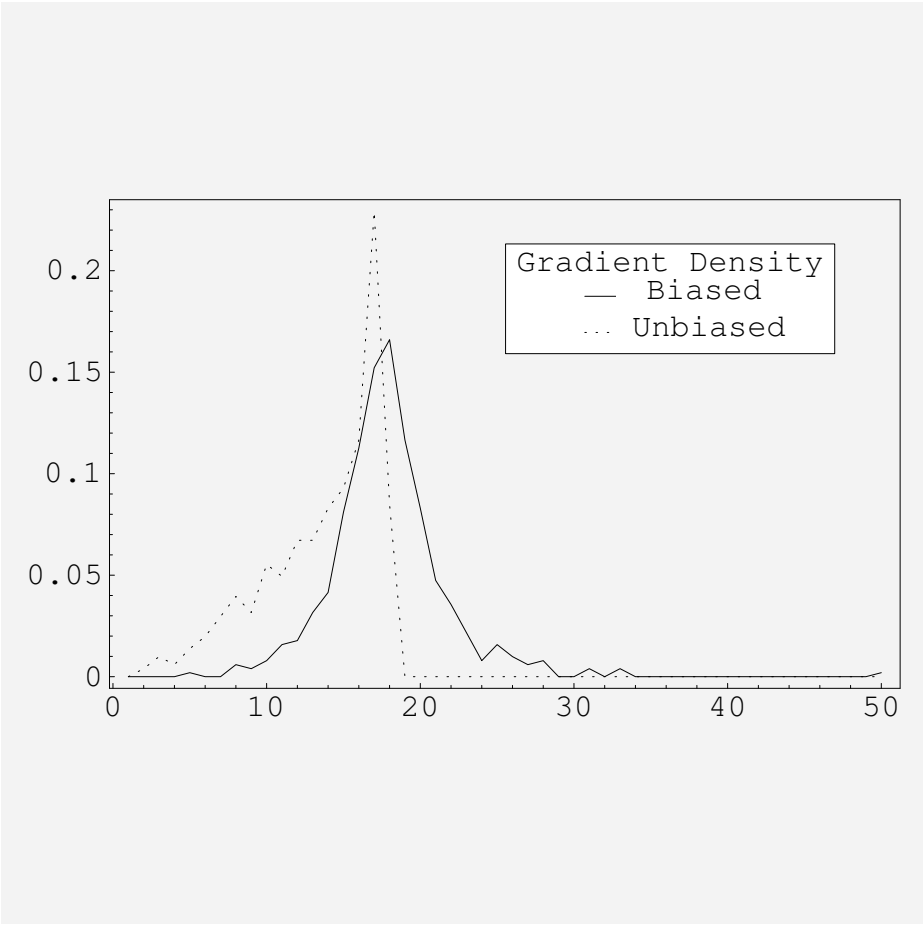

Fig. 29. Gradient densities for the first parameter surface: case 2

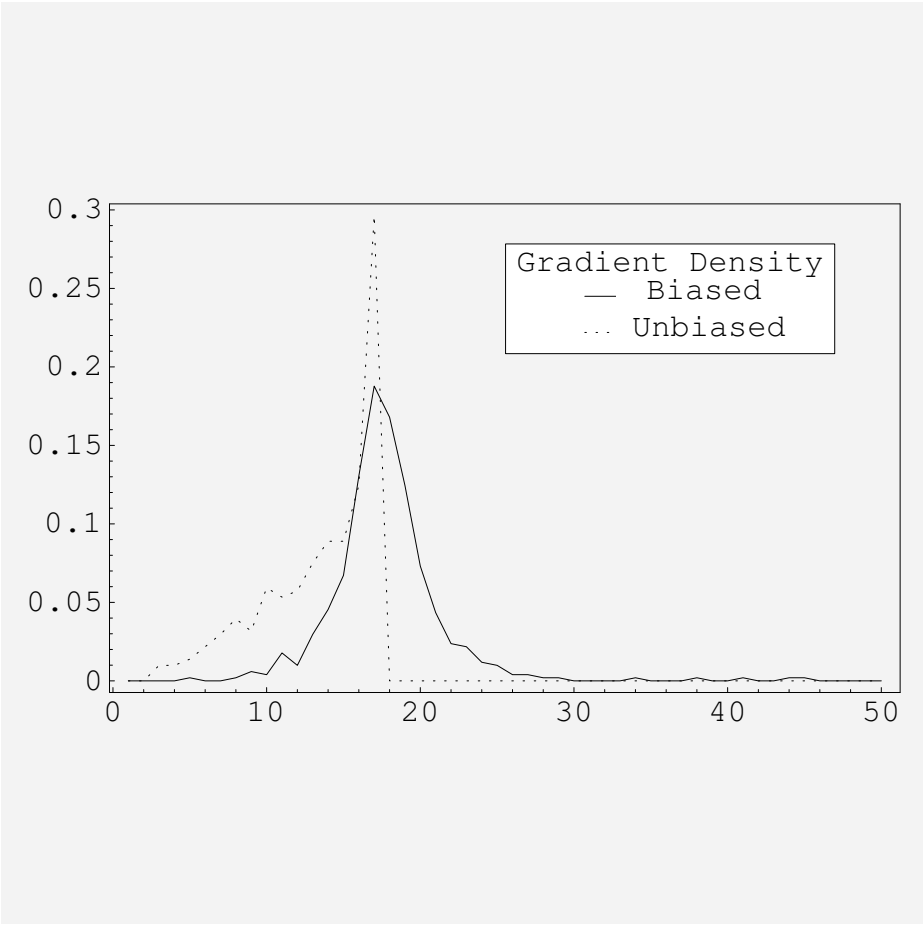

Fig. 30. Gradient densities for the first parameter surface: case 3

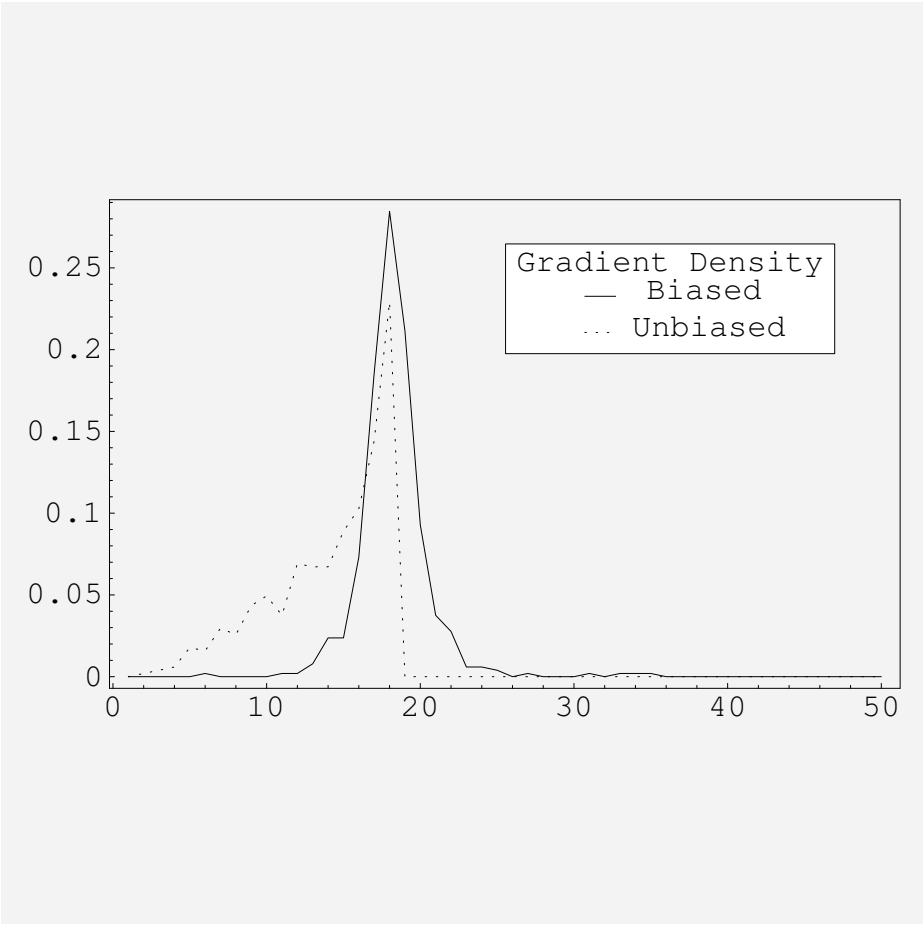

Fig. 31. Gradient densities for the first parameter surface: case 4

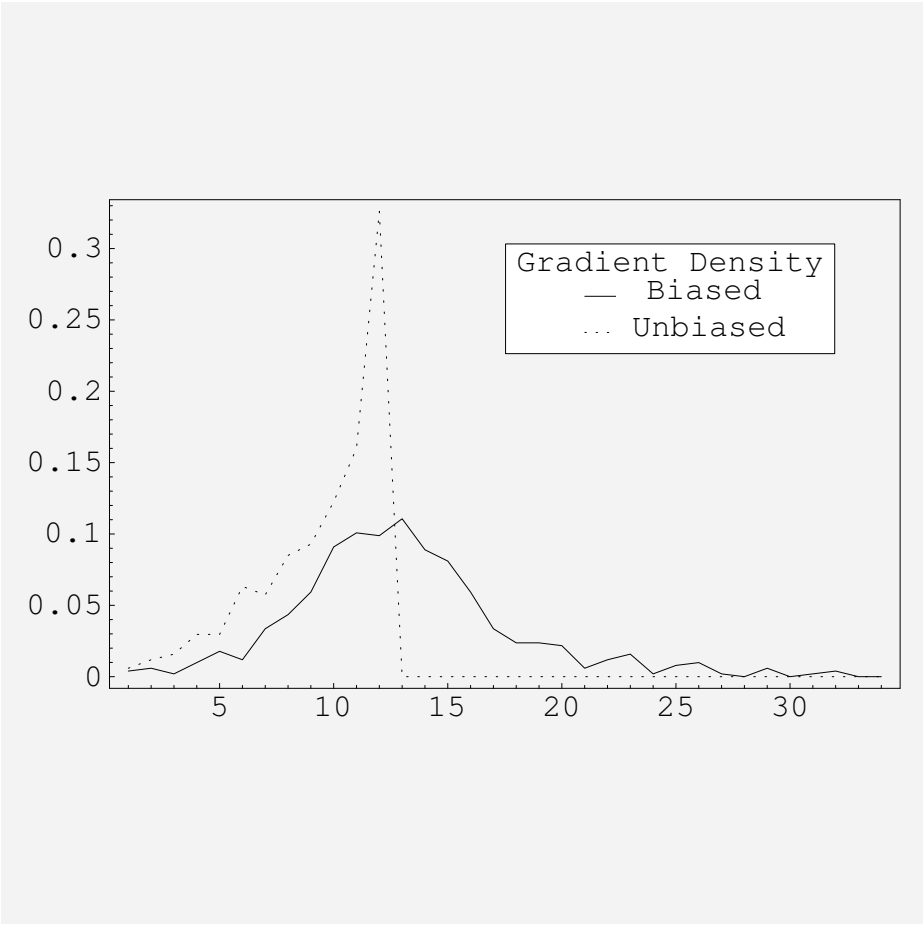

Fig. 32. Gradient densities for the first parameter surface: case 5

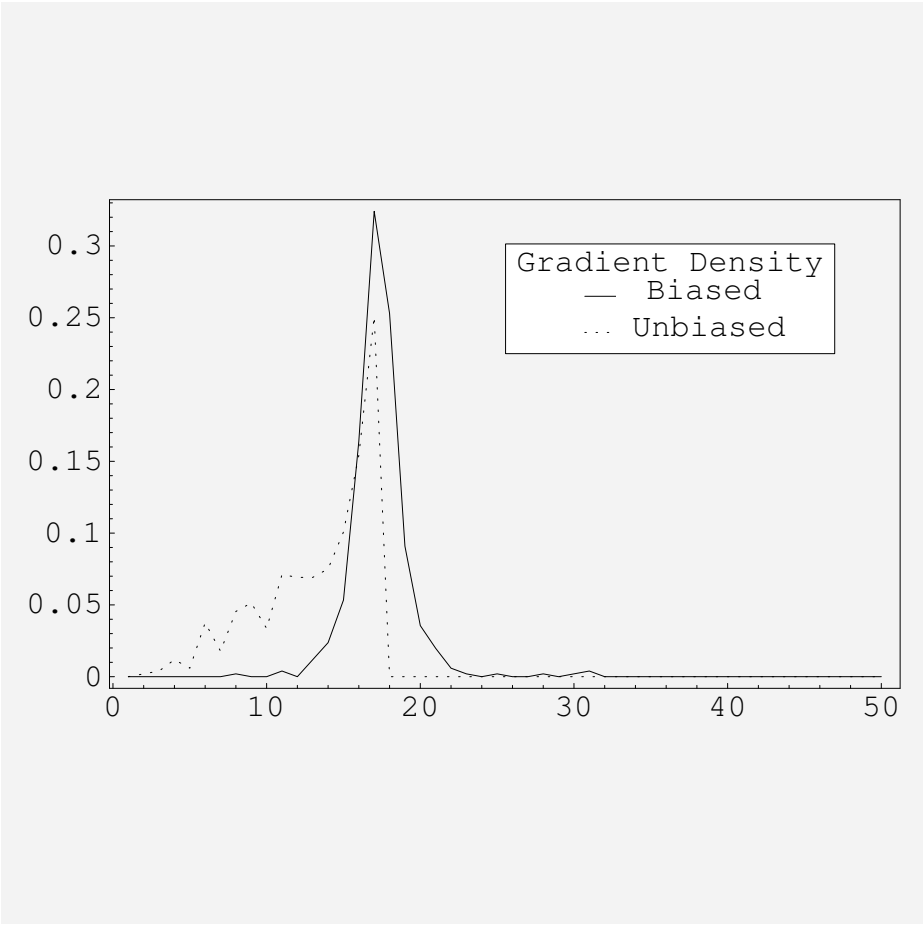

Fig. 33. Gradient densities for the first parameter surface: case 6

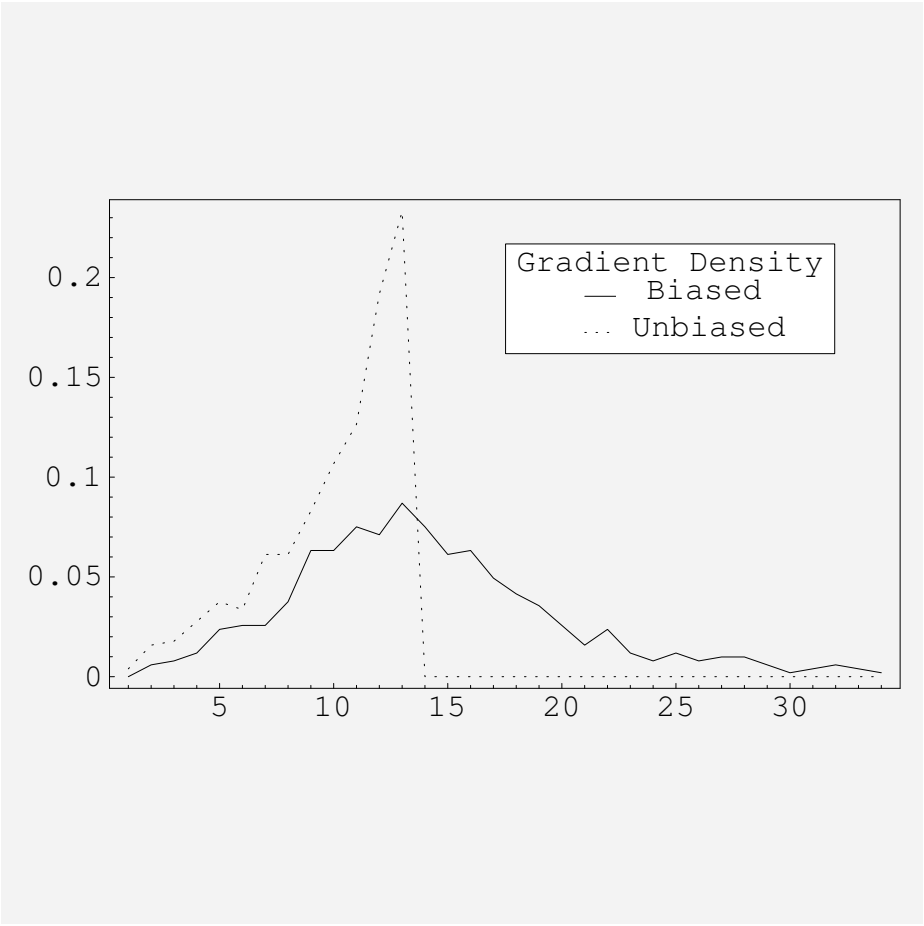

Fig. 34. Gradient densities for the first parameter surface: case 7

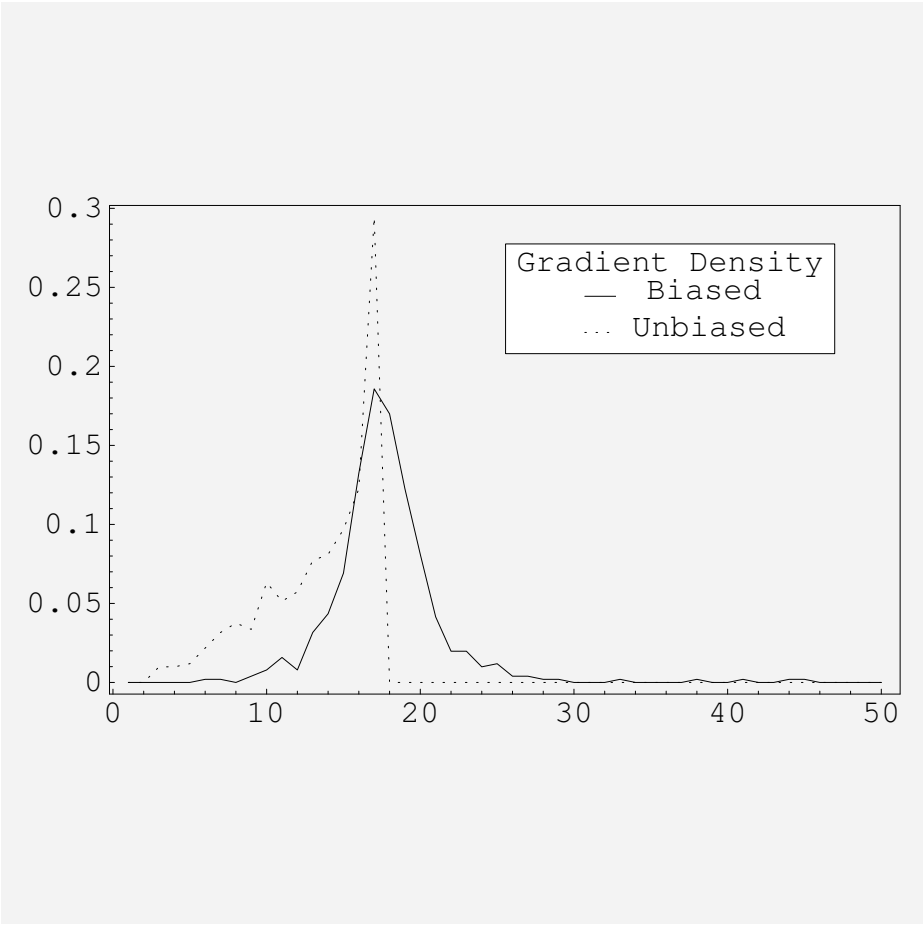

Fig. 35. Gradient densities for the first parameter surface: case 8

Table II. Means for each case scenario for the 1st parameter surface

Case	Mean	
	Biased	Unbiased
1	1.03695	0.783702
2	1.04196	0.789037
3	1.03747	0.783448
4	1.06531	0.81791
5	1.17332	0.816654
6	1.01128	0.774322
7	1.31643	0.856068
8	1.03695	0.783702

3. Parameter Surface Example 2

In this example a new parameter surface is used. This time, the correlation $R_{xx}[2]$ is scaled to geometric mean of correlation's $R_{xx}[0]$ and $R_{xx}[1]$. We have now:

$$c(a,b) = \lambda\sqrt{ab} \tag{3.51}$$

where λ is the scaling coefficient that takes a value between $-1 < \lambda < 1$. Note that the previous equation represents open parameter surface. Such a model might be appropriate in situations where one had much more confidence in the values of the correlations $R_{xx}[0]$ and $R_{xx}[1]$. Various λ can be considered.

For a fixed and known value of λ, we will need to compute the equalizing factor.

$$Equalizing\ factor \ = \ \sqrt{1 + (\frac{\partial c}{\partial a})^2 + (\frac{\partial c}{\partial b})^2}$$

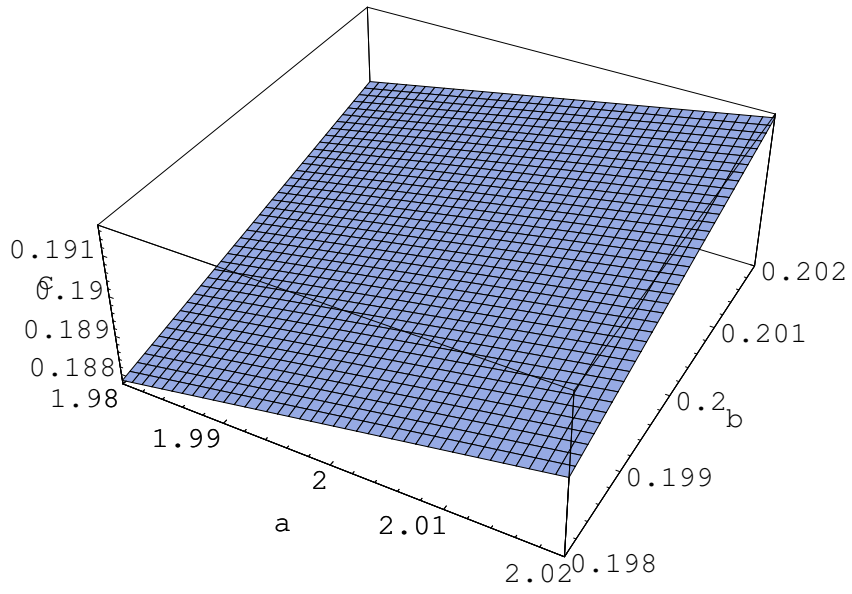

Fig. 36. Representation of the open surface for $\lambda = 0.3$

The parameter surface for the one value of λ is represented in figure 36 and performance surface in figure 37.

4. Simulation Results

Like the previous section, we determine sets of points (a, b, c) that respect the equation of the open surface. In order to accomplish this, we partition the (a, b) plane using squares out as far as possible (the open surface has no boundaries). One of the requirements for all those sample densities is the need for positive definiteness for all sets of points. Each corresponding C matrix formed with those points has the particularity of being real and symmetrical forming an Hermitian matrix. The Hermitian matrix has special properties that could be used by one to verify that the sets of points are positive definite. One would only need to compute all the eigenvalues, knowing that if all the eigenvalues are positive, then the matrix is positive definite. Note that any not-positive definite sets of points are dropped by the simulation program.

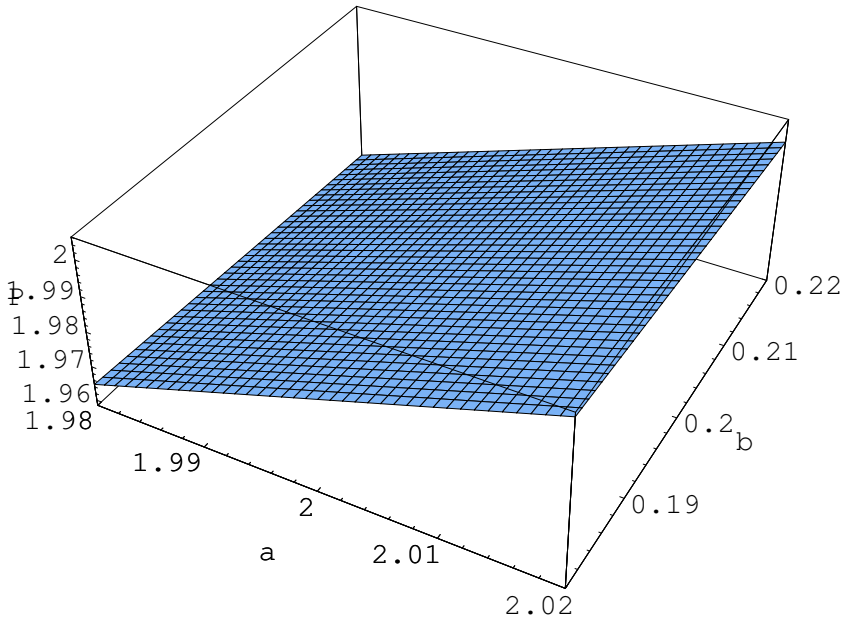

Fig. 37. Representation of the performance surface for open parameter surface

For each case scenario, (see table III) the focus of our interest is on both biased and unbiased, cases. All the gradient distributions are based on 1200 points each. For all the densities the bin-size is $5 * 10^{-4}$, over range of $1.015 - 1.065$, this generates 100 bins. There are six sets of two graphs that cover all those cases (see figures 38 to 43).

By varying the nominal values, one will be able to visualize the robustness of the estimation scheme. The changes will be made one value at the time, to improve the chance of visualizing the parameter's effect on the robustness of the estimation scheme. Table IV gives the corresponding means for both the biased and unbiased case, for each case scenario.

Table III. All the different case scenarios for the 2nd surface parameter

Candidate	λ	a_0	b_0	k_1	k_2
1	0.5	2	0.2	0.10734	0.0631206
2	-0.5	2	0.2	-0.11739	0.0790114
3	0.5	2	0.5	0.150221	0.150221
4	-0.5	2	0.5	-0.214737	0.214737
5	0.5	4	0.5	0.11811	0.0779488
6	-0.5	4	0.5	-0.133858	0.10022

Table IV. Means for each case scenario for the 2nd parameter surface

Case	Mean	
	Biased	Unbiased
1	1.1759	1.17021
2	1.03894	1.03444
3	1.05818	1.05802
4	1.1759	1.17021
5	1.02808	1.02673
6	1.05329	1.04876

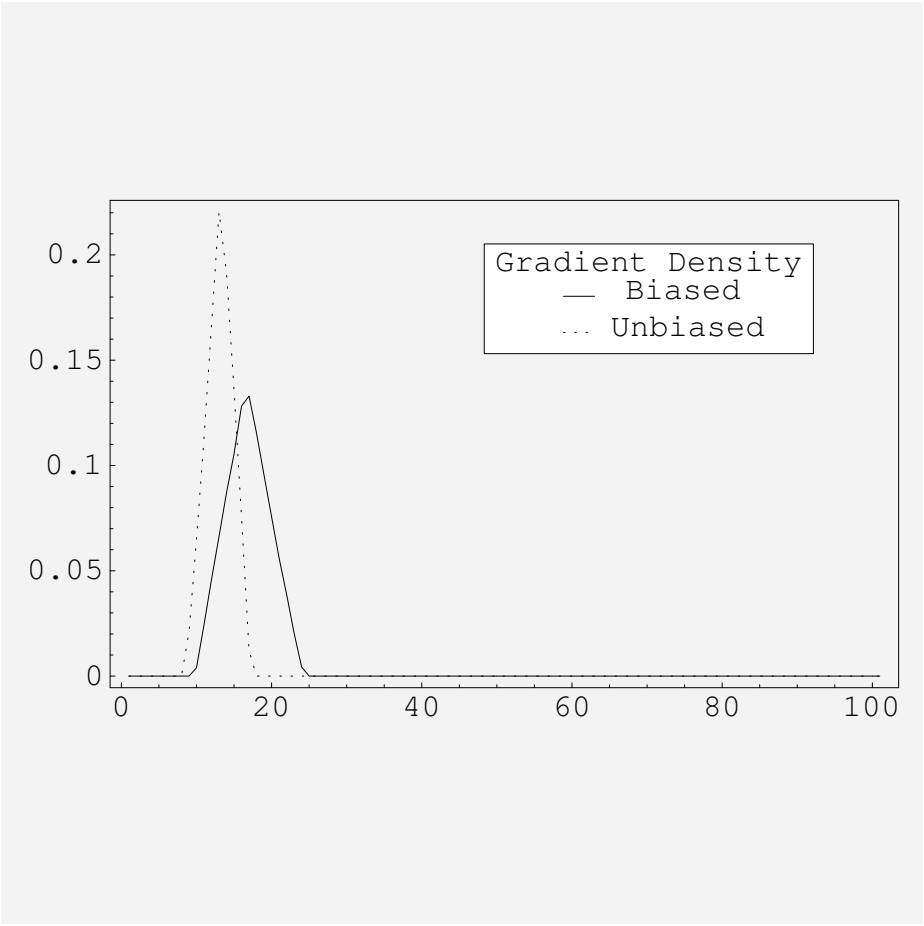

Fig. 38. Gradient densities for the second parameter surface: case 1

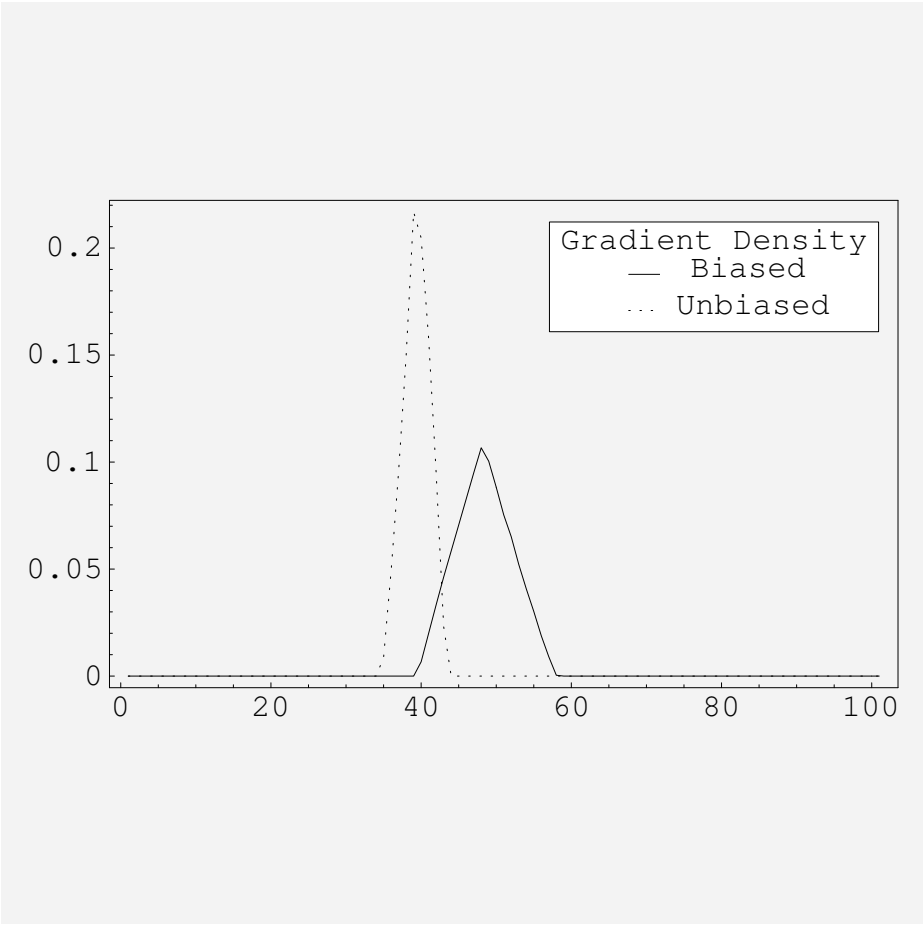

Fig. 39. Gradient densities for the second parameter surface: case 2

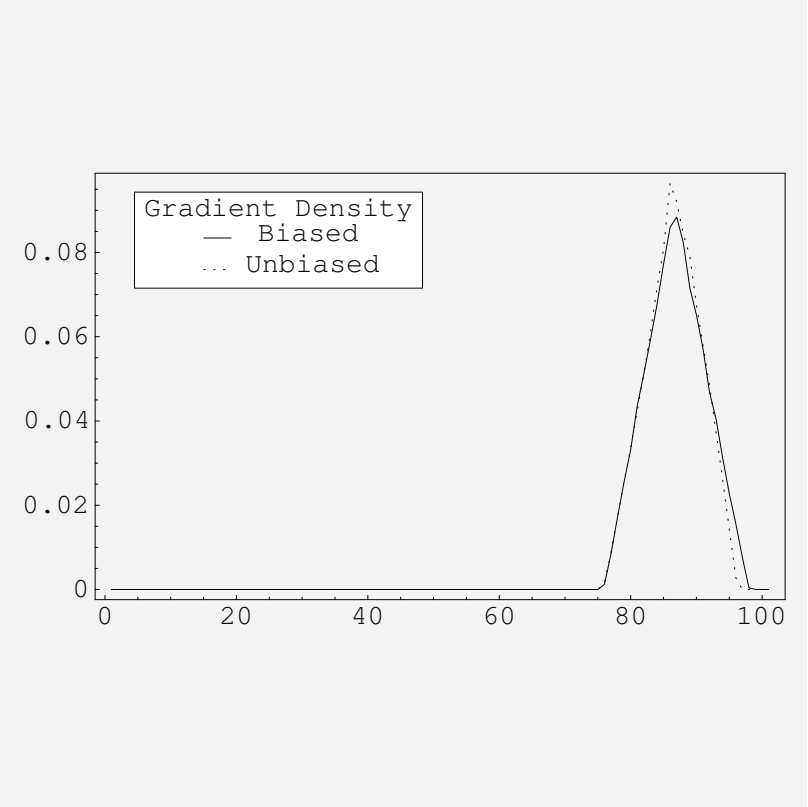

Fig. 40. Gradient densities for the second parameter surface: case 3

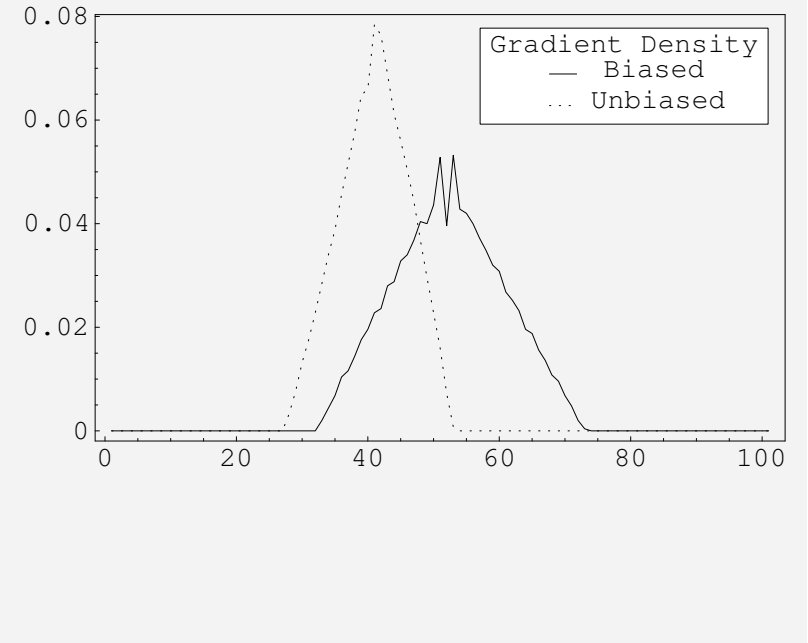

Fig. 41. Gradient densities for the second parameter surface: case 4

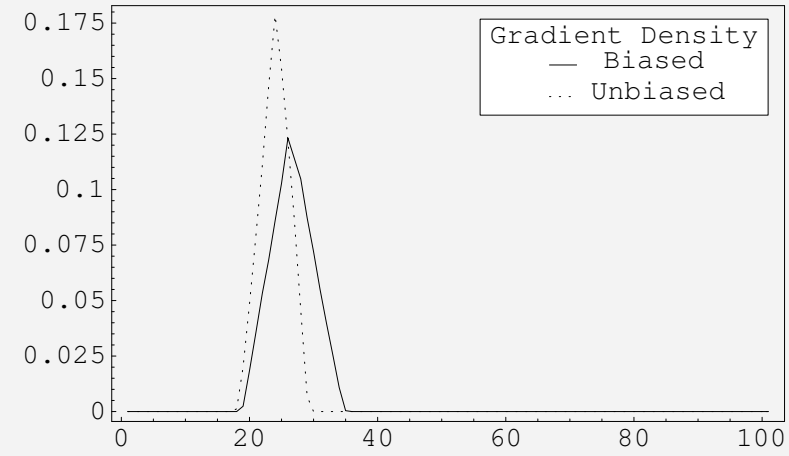

Fig. 42. Gradient densities for the second parameter surface: case 5

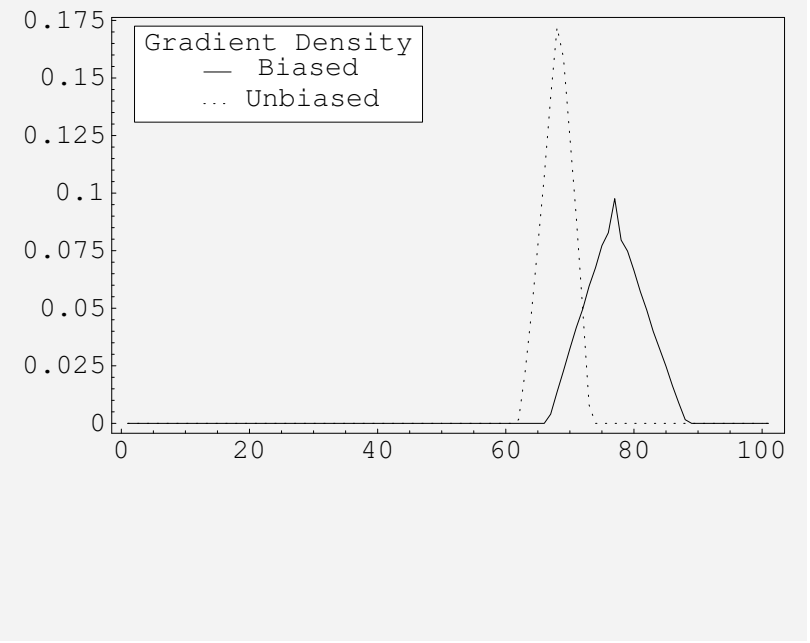

Fig. 43. Gradient densities for the second parameter surface: case 6

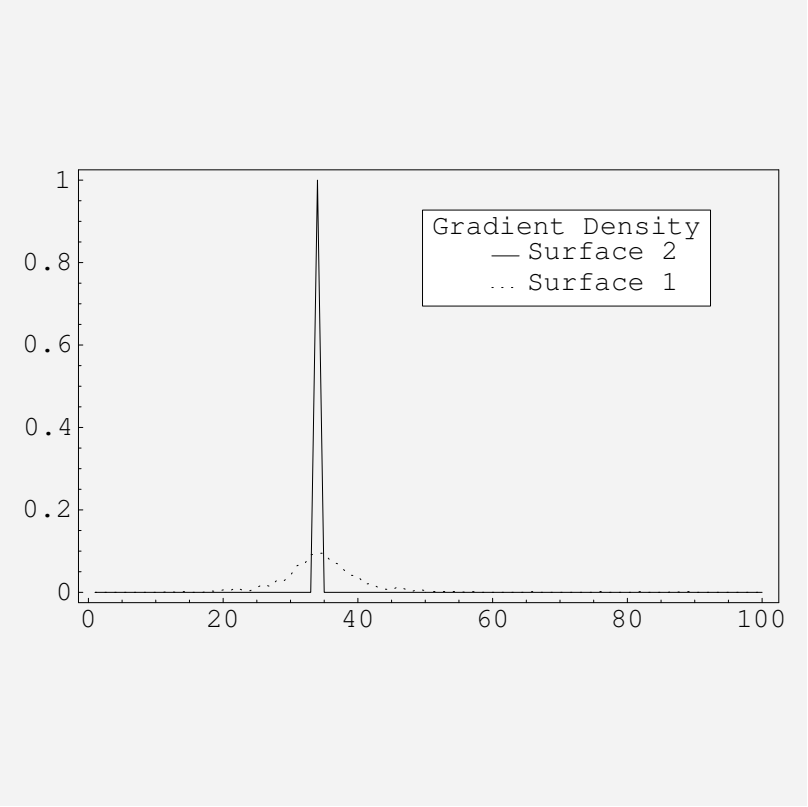

Fig. 44. Gradient densities for the first and second parameter surface (biased)

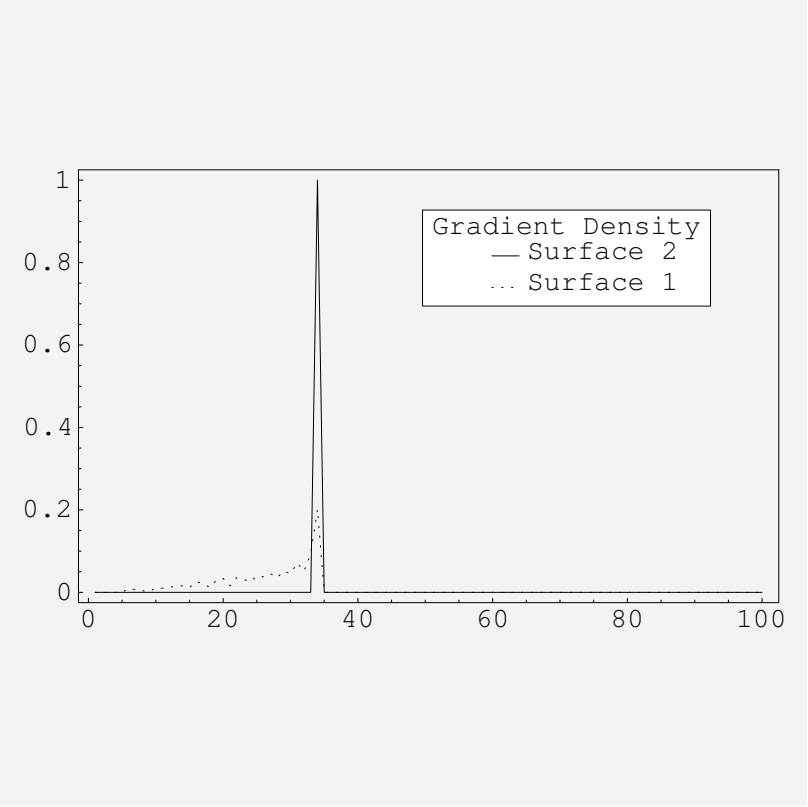

Fig. 45. Gradient densities for the first and second parameter surface (unbiased)

H. Conclusion

Distributions/probability densities of the gradient over the performance manifold are used to visualize if the estimation algorithm is or is not robust. The more "compactness" seen in the form factor of the density curve, the more it shows the absence of big slopes, and the less variable is the gradient. If in addition the means of the gradient distribution is of small value, then the estimator might be called "robust".

In the case of the "closed" ellipsoid shape which corresponds to a closed parameter surface with a positive curvature, the Gradient densities obtained for the unbiased case are showing a high degree of robustness (absence of large slopes) as compared to the biased model. The means look very consistent with Gradient densities and are consistently lower for the unbiased model. This implies that the robustness of the system is higher than seen in biased model, also implying the manifold is distorted to a higher degree under bias.

In the case of the "open" surface, which this time corresponds to an open parameter surface with a negative curvature, the Gradient densities obtained show a presence of robustness in both schemes. The means under unbiased model are lower, however the decreases in mean isn't a huge, implying that manifold isn't so distorted under bias. This makes one alternatively use biased model, which is less complex in computational intensity.

To the view of those results we can conclude that the statement about robustness of an algorithm can only be made if it is associated with a parameter surface. The results depend completely on the choice of the parameter surface. Figures 44 and 45 represents comparison of robustness for similar nominals, both surface 1 and 2 superimposed, illustrating surface 2 is more robust than surface 1.

Since the unbiased approach is preferable in terms of telling the "true" robust-

ness story, our results show that the extra difficulty in computing unbiased gradient densities is worth the effort in the simulations domain. Nevertheless, we remark that unbiased approach involves a matrix inversion, G^{-1}, linear algebra operation. The complexity of matrix inversion in hardware becomes prohibitive for real time applications and large values of matrix dimension. This operation is inherently complex to design in hardware domain, more so in real time systems and requires fast, pipelined and scalable hardware architecture. Often iterative algorithms are used in such situations. Bridging the gap between these two highly specialized domains allows more degrees of freedom when determining the final solution. This however is beyond the scope of this dissertation.

The next step of this work is an extension of our research: the case where non-stationarity is involved. While the extension of this work to non-stationary is not trivial, nevertheless, it will allow us to draw conclusions regarding the effect of non-stationarity on robustness.

CHAPTER IV

ROBUSTNESS MODELS FOR NON-STATIONARY SYSTEMS

Algorithms used in telecommunications, image, speech, radar signal processing and econometrics rely on various degrees of measuring performance. The measure of robustness, is also based on the received data; for example the sampling speed has a direct effect on the stationarity of the data. We have seen that the Huber Strassen method resists admitting non-stationarity and dependency. We note that in engineering there is an important class of applications where non stationarity can be admitted through the use of higher dimensionality. For example, covariance and correlation matrices arise quite often, and the differences between the stationary and non stationary case is the dimension of the matrix. We propose the admission of non stationarity for the class of applications where the uncertain knowledge can be still modeled as a manifold but of higher dimension than for the stationary case. This chapter starts by extending the previous example of three dimensions to six dimensions, to develop the model for a non-stationary system. It then introduces the necessary computations one needs to make in order to state on the measure of robustness.

A. Mathematical Model for Non-Stationary Systems

In the mathematical sciences, a stationary process (or strictly stationary process) is a stochastic process whose probability distribution at a fixed time or position is the same for all times or positions. As a result, parameters such as the mean and variance, if they exist, also do not change over time or position. Processes whose statistical properties do change are referred to as non stationary.

As an example, white noise is stationary. However, the sound of a cymbal crashing is not stationary because the acoustic power of the crash (and hence its variance)

diminishes with time. The measurement of white noise is considered a stationary process. Alternatively, the measurement of a slow sampling process is not stationary. The underwater communications channel is non-stationary because of multiple reflections of sound waves off the bottom and moving water surface and relative transmitter and receiver movement.

1. The Linear Prediction Filter

The linear prediction of a random variable $X[3] \equiv X_3$ in terms of $X[2] \equiv X_2$ and $X[1] \equiv X_1$ is:

$$\hat{X}[3] = k_1 X[1] + k_2 X[2] \tag{4.1}$$

where X[.] are Gaussian distributed random variables with covariance matrix C_N:

$$C_N = \begin{pmatrix} E[X_1^2] & E[X_1X_2] & E[X_1X_3] \\ E[X_1X_2] & E[X_2^2] & E[X_2X_3] \\ E[X_1X_3] & E[X_2X_3] & E[X_3^2] \end{pmatrix}$$

where $X_1 \sim \mathcal{N}(0, a)$, $X_2 \sim \mathcal{N}(0, b)$ and $X_3 \sim \mathcal{N}(0, c)$. Thus,

$$C_N = \begin{pmatrix} a & d/\sqrt{2} & f/\sqrt{2} \\ d/\sqrt{2} & b & e/\sqrt{2} \\ f/\sqrt{2} & e/\sqrt{2} & c \end{pmatrix}$$

and $E[X_1X_2] = d/\sqrt{2}$, $E[X_1X_3] = f/\sqrt{2}$, $E[X_2X_3] = f/\sqrt{2}$. The normalization by $\sqrt{2}$ is done to make the higher dimensional analysis easier; it does not compromise the robustness analysis.

2. The Parameter Surface

The parameter surface obtained using the Frobenius norm, for the non-stationary case, would be a 5-sphere in 6D:

$$\left\| \begin{pmatrix} a & d/\sqrt{2} & f/\sqrt{2} \\ d/\sqrt{2} & b & e/\sqrt{2} \\ f/\sqrt{2} & e/\sqrt{2} & c \end{pmatrix} - \begin{pmatrix} a_0 & d_0/\sqrt{2} & f_0/\sqrt{2} \\ d_0/\sqrt{2} & b_0 & e/\sqrt{2} \\ f_0/\sqrt{2} & e_0/\sqrt{2} & c_0 \end{pmatrix} \right\| = \|E_N\| = \epsilon_N \quad (4.2)$$

For this example ($3 \star 3$ matrix), let $\|E_N\|$ represent the value of the *Frobenius Norm.* Then:

$$\|E_N\|^2 = (a-a_0)^2 + (b-b_0)^2 + (c-c_0)^2 + (d-d_0)^2 + (e-e_0)^2 + (f-f_0)^2 = {\epsilon_N}^2 \quad (4.3)$$

Thus,

$$f = f_0 \pm \sqrt{{\epsilon_N}^2 - (a-a_0)^2 + (b-b_0)^2 + (c-c_0)^2 + (d-d_0)^2 + (e-e_0)^2} \quad (4.4)$$

Thus,

$$f(a,b,c,d,e) = f_0 \pm \sqrt{{\epsilon_N}^2 - (a-a_0)^2 + (b-b_0)^2 + (c-c_0)^2 + (d-d_0)^2 + (e-e_0)^2} \quad (4.5)$$

3. The MSE Performance Criterion

The forward prediction error is denoted as

$$e_n = \hat{X}[n] - X[n] \quad (4.6)$$

The Mean Square Error is defined as,

$$MSE = E[e[n]^2] \quad (4.7)$$

Therefore the MSE Performance Criterion becomes:

$$\begin{aligned} MSE &= E[[\hat{X}_3 - X_3]^2] \\ &= E[[k_1X_1 + k_2X_2 - X_3]^2] \end{aligned} \tag{4.8}$$

After few computations, the test statistic can easily be transformed into:

$$\begin{aligned} MSE &= E[X_3]^2 + k_1^2E[X_1]^2 + k_2^2E[X_2]^2 \\ &- 2k_1E[X_1X_3] - 2k_2E[X_2X_3] + 2k_1k_2E[X_1X_2] \end{aligned} \tag{4.9}$$

Thus the MSE depends on the correlations of the random variables X_3, X_2 and X_1. For the non-stationary case MSE is equal to:

$$\begin{aligned} MSE\Big|_{NS} &= c + k_1^2a + k_2^2b \\ &- 2k_1f/\sqrt{2} - 2k_2e/\sqrt{2} + 2k_1k_2d/\sqrt{2} \end{aligned} \tag{4.10}$$

Using equation 4.5

$$\begin{aligned} MSE\Big|_{NS} &= c + k_1^2a + k_2^2b \\ &- 2k_1f(a,b,c,d,e)/\sqrt{2} - 2k_2e/\sqrt{2} + 2k_1k_2d/\sqrt{2} \end{aligned} \tag{4.11}$$

$$MSE\Big|_{NS} = P(a,b,c,d,e) \tag{4.12}$$

For the computation of the value of the coefficients k_1, k_2, one will employ nominal values, chosen according to the table V

B. Equalizing Factors and Point Selection

For the non-stationary case, the point assignment and the weighting factor are done simultaneously (for a sphere in 6D). One will first start with a sphere in 3D and then expand our reasoning to a sphere in 6D. For a sphere in 3D, centered at the origin,

Table V. Nominal values for the non-stationary case

Cor-relation	Nominal Value
$E[X_1^2]$	a_0
$E[X_2^2]$	b_0
$E[X_3^2]$	c_0
$E[X_1X_2]$	$d_0/\sqrt{2}$
$E[X_2X_3]$	$e_0/\sqrt{2}$
$E[X_1X_3]$	$f_0/\sqrt{2}$

one has (see figures 46 and 47) a radius in the (x_1, x_2) plane of $\sqrt{\epsilon}\cos\phi_2$. Hence:

$$\begin{aligned} x_1 &= (\cos\phi_1)\sqrt{\epsilon}\cos\phi_2 = \sqrt{\epsilon}\cos\phi_2\cos\phi_1 \\ x_2 &= (\sin\phi_1)\sqrt{\epsilon}\cos\phi_2 = \sqrt{\epsilon}\cos\phi_2\sin\phi_1 \\ x_3 &= \sqrt{\epsilon}\sin\phi_2 \end{aligned} \tag{4.13}$$

where $0 \leq \phi_1 < 2\pi$ and $-\pi/2 \leq \phi_2 \leq \pi/2$.

The equalizing factor is then equal to the surface area element for the sphere in polar coordinates (see figure 47). As ϕ_2 is incremented by $d\phi_2$, it introduces an increment in latitude of $\sqrt{\epsilon}d\phi_2$. As ϕ_1 is incremented, it induces an increment in longitude of $(\sqrt{\epsilon}\cos\phi_2)d\phi_1$. After noticing that all the increments are orthogonal, the surface area element is equals to:

$$\begin{aligned} Surface\ area\ element &= \sqrt{\epsilon}d\phi_2\sqrt{\epsilon}\cos\phi_2 d\phi_1 \\ &= \epsilon\cos\phi_2 d\phi_1 d\phi_2 \end{aligned} \tag{4.14}$$

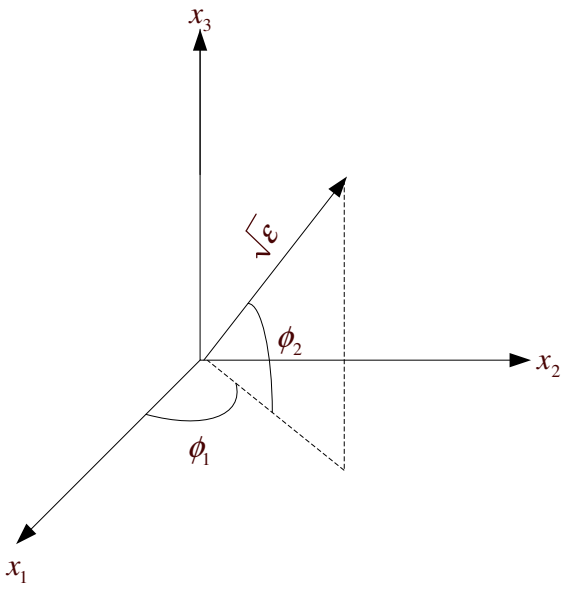

Fig. 46. Selection of the sets of points for the non-stationary case

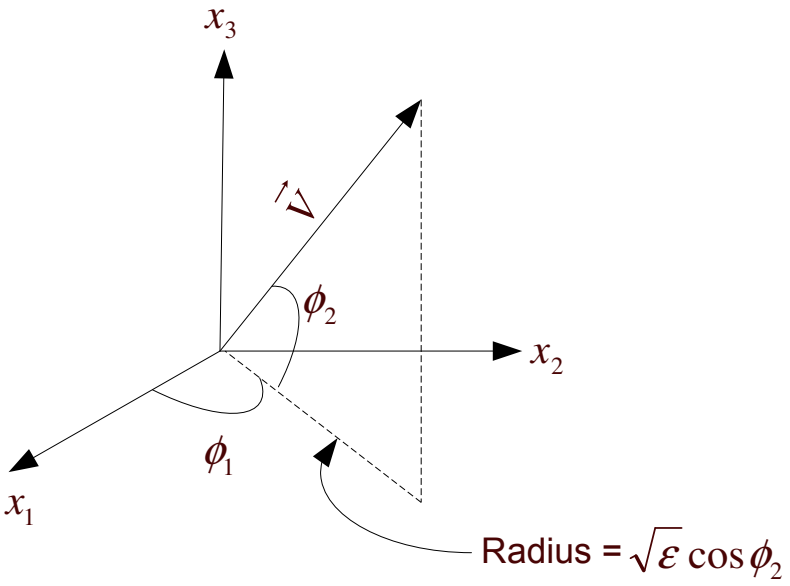

Fig. 47. The selection of the sets of points using polar coordinates for the non-stationary case

The equalizing factor for the sphere in 3D then becomes:

$$equalizing\ factor\ 3D = \epsilon|\cos\phi_2| \tag{4.15}$$

Now, one needs to expand the previous approach to a sphere in 4D. Therefore, one will need to project down into the (x_1, x_2, x_3) space, where the vector $\overrightarrow{V}$ is at the angle in 4D of ϕ_3 with respect to the (x_1, x_2, x_3) space. Thus the "radius" in (x_1, x_2, x_3) is no longer $\sqrt{\epsilon}$ but $\sqrt{\epsilon}\cos\phi_3$. One hence obtain:

$$\begin{aligned} x_1 &= (\sqrt{\epsilon}\cos\phi_3)\cos\phi_2\cos\phi_1 = \sqrt{\epsilon}\cos\phi_3\cos\phi_2\cos\phi_1 \\ x_2 &= \sqrt{\epsilon}\cos\phi_3\cos\phi_2\sin\phi_1 \\ x_3 &= \sqrt{\epsilon}\cos\phi_3\sin\phi_2 \\ x_4 &= \sqrt{\epsilon}\sin\phi_3 \end{aligned} \tag{4.16}$$

where $0 \leq \phi_1 < 2\pi$, $-\pi/2 \leq \phi_i \leq \pi/2$, for i = 2,3. Now, the surface area element is equal to a surface area element of a sphere in 3D with radius $\sqrt{\epsilon}\cos\phi_3$ times $\sqrt{\epsilon}d\phi_3$. Hence:

$$\begin{aligned} Surface\ area\ element &= (\sqrt{\epsilon}d\phi_3)(\sqrt{\epsilon}\cos\phi_3 d\phi_2)(\sqrt{\epsilon}\cos\phi_3\cos\phi_2 d\phi_1) \\ &= \varepsilon^{3/2}\cos^2\phi_3\cos\phi_2 d\phi_1 d\phi_2 d\phi_3 \end{aligned} \tag{4.17}$$

The equalizing factor for the 4D sphere then becomes:

$$equalizing\ factor\ 4D = \epsilon^{3/2}\cos^2\phi_3|\cos\phi_2| \tag{4.18}$$

Now, with inductive reasoning, for a sphere in 5D we have:

$$\begin{aligned} x_1 &= \sqrt{\epsilon}\cos\phi_4\cos\phi_3\cos\phi_2\cos\phi_1 \\ x_2 &= \sqrt{\epsilon}\cos\phi_4\cos\phi_3\cos\phi_2\sin\phi_1 \\ x_3 &= \sqrt{\epsilon}\cos\phi_4\cos\phi_3\sin\phi_2 \\ x_4 &= \sqrt{\epsilon}\cos\phi_4\sin\phi_3 \\ x_5 &= \sqrt{\epsilon}\sin\phi_4 \end{aligned} \tag{4.19}$$

where $0 \leq \phi_1 \leq 2\pi$ and $-\pi/2 \leq \phi_i \leq \pi/2$, i = 2,3,4. The surface area element is then equal to:

$$\begin{aligned} Surface\ area\ element &= \sqrt{\epsilon}d\phi_4(surface\ area\ element\ of \\ &\qquad 4D\ sphere\ with\ radius\ \sqrt{\epsilon}\cos\phi_4) \\ &= \sqrt{\epsilon}d\phi_4(\sqrt{\epsilon}\cos\phi_4)^3\cos^2\phi_3|\cos\phi_2|d\phi_1 d\phi_2 d\phi_3 \end{aligned} \tag{4.20}$$

The equalizing factor of a sphere in 5D is then equal to:

$$equalizing\ factor\ 5D = \epsilon^2|\cos^3\phi_4\cos^2\phi_3\cos\phi_2| \tag{4.21}$$

The same reasoning is applied to a sphere in 6D. We then obtain:

$$\begin{aligned} x_1 &= \sqrt{\epsilon}\cos\phi_5\cos\phi_4\cos\phi_3\cos\phi_2\cos\phi_1 \\ x_2 &= \sqrt{\epsilon}\cos\phi_5\cos\phi_4\cos\phi_3\cos\phi_2\sin\phi_1 \\ x_3 &= \sqrt{\epsilon}\cos\phi_5\cos\phi_4\cos\phi_3\sin\phi_2 \\ x_4 &= \sqrt{\epsilon}\cos\phi_5\cos\phi_4\sin\phi_3 \\ x_5 &= \sqrt{\epsilon}\cos\phi_5\sin\phi_4 \\ x_6 &= \sqrt{\epsilon}\sin\phi_5 \end{aligned} \tag{4.22}$$

where $0 \leq \phi_1 \leq 2\pi$ and $-\pi/2 \leq \phi_i \leq \pi/2$, i = 2,3,4,5. The equalizing factor of a sphere in 6D is then equal to:

$$Equalizing\ factor\ 6D = \epsilon^{5/2} |\cos^4 \phi_5 \cos^3 \phi_4 \cos^2 \phi_3 \cos \phi_2| \tag{4.23}$$

1. Transforming Equalization to Cartesian Co-Ordinates

We leverage a theorem [61, 62] which effectively describes how lengths, areas, volumes, and generalized n-dimensional volumes (contents) are distorted by differentiable functions. In particular, the change of variables theorem reduces the whole problem of figuring out the distortion of the content to understanding the infinitesimal distortion, i.e., the distortion of the derivative (a linear map), which is given by the linear map's determinant. So $f : R^n \rightarrow R^n$is an area-preserving linear transformation iff $|det(f') = 1|$, and in more generality, if S is any subset of R^n, the content of its image is given by $|det(f')|$ times the content of the original. The change of variables theorem takes this infinitesimal knowledge, and applies calculus by breaking up the domain into small pieces and adds up the change in area, bit by bit. The change of variables theorem is a simple consequence of the curl theorem. The generalization to n dimensions requires no additional assumptions other than the regularity conditions on the boundary.

$$dy_1 \cdots dy_n = \left| \frac{\partial y_1, \cdots, \partial y_n}{\partial x_1, \cdots, \partial x_n} \right| dx_1 \cdots dx_n \tag{4.24}$$

The Jacobian matrix, sometimes simply called "the Jacobian"

$$J(y_1, \cdots, y_n) = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_n}{\partial x_1} & \cdots & \frac{\partial y_n}{\partial x_n} \end{pmatrix} \tag{4.25}$$

The determinant of J is the Jacobian determinant and is denoted:

$$J = \Big|\frac{\partial y_1, \cdots, \partial y_n}{\partial x_1, \cdots, \partial x_n}\Big| \tag{4.26}$$

Applying this to equation 4.23

$$\begin{aligned} dx_1dx_2dx_3dx_4dx_5 &= \Big|\frac{\partial x_1, \cdots, \partial x_5}{\partial \phi_1, \cdots, \partial \phi_5}\Big| d\phi_1 d\phi_2 d\phi_3 d\phi_4 d\phi_5 \\ &= \epsilon^{5/2} |\cos^4\phi_5 \cos^3\phi_4 \cos^2\phi_3 \cos\phi_2 \sin\phi_5| d\phi_1 d\phi_2 d\phi_3 d\phi_4 d\phi_5 \end{aligned} \tag{4.27}$$

Thus, in Cartesian co-ordinates:

$$Equalizing\ factor\ 6D = \frac{1}{|\sin\phi_5|} \tag{4.28}$$

$$= \frac{\sqrt{\epsilon}}{x_6} \tag{4.29}$$

C. Directional Derivatives

We have:

$$\Big(D_{\vec{X}}h\Big)\Big|_{Extreme} = \sqrt{\nabla h G^{-1} \nabla h^T} \tag{4.30}$$

The computations of the directional derivative $(D_{\vec{X}}h) = \sqrt{\nabla h G^{-1} \nabla h^T}$ depends directly on the approach used, but does not depend on the choice of the coordinates (provided the underlying manifold remains fixed). However, this follows from the classical interpretation of directional derivative as a limit of ∇h over the arc length, where the answer computes to be independent of the curve chosen as long as the tangent to the curve at P is fixed.

1. Biased Perturbations-Suboptimal Model

In the case where the slope is biased with perturbations, the variations are translated to the Euclidean space. It is with these variations that the bias enters the system, because the manifold has been altered. Thus, $g_{ij} = \delta_{ij}$, where δ_{ij} is the *Kronecker delta*. This time we have:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

G becomes the identity matrix. If the performance function is $P = h(a, b, c, d, e, f) \equiv h(a, b, c, d, e)$, we then obtain:

$$\nabla h = (\frac{\partial P}{\partial a}, \frac{\partial P}{\partial b}, \ldots, \frac{\partial P}{\partial e}) \tag{4.31}$$

Using the new matrix G and ∇h into equation 4.30, we obtain:

$$\begin{aligned} Slope\Big|_{Biased} &= \left(D_{\overline{X}} h\right)\Big|_{Extreme} && (4.32) \\ &= \sqrt{\nabla h G^{-1} \nabla h^T} && (4.33) \\ &= \sqrt{(\frac{\partial h}{\partial a})^2 + (\frac{\partial h}{\partial b})^2 + (\frac{\partial h}{\partial c})^2 + (\frac{\partial h}{\partial d})^2 + (\frac{\partial h}{\partial e})^2} && (4.34) \end{aligned}$$

Thus,

$$Slope\Big|_{Biased} = \sqrt{(\frac{\partial h}{\partial a})^2 + (\frac{\partial h}{\partial b})^2 + (\frac{\partial h}{\partial c})^2 + (\frac{\partial h}{\partial d})^2 + (\frac{\partial h}{\partial e})^2} \tag{4.35}$$

The equation (4.35) is the well known version of the directional derivative applying on functions on affine space.

2. Unbiased Perturbations-Optimal Model

The gradient of six variables embedded into 5D becomes (the parameter surface is defined as $f(a, b, c, d, e)$):

$$\frac{\partial}{\partial a} = (1, 0, 0, 0, 0, \frac{\partial f}{\partial a}) \tag{4.36}$$
$$\frac{\partial}{\partial b} = (0, 1, 0, 0, 0, \frac{\partial f}{\partial b}) \tag{4.37}$$
$$\frac{\partial}{\partial c} = (0, 0, 1, 0, 0, \frac{\partial f}{\partial c}) \tag{4.38}$$
$$\frac{\partial}{\partial d} = (0, 0, 0, 1, 0, \frac{\partial f}{\partial d}) \tag{4.39}$$
$$\frac{\partial}{\partial e} = (0, 0, 0, 0, 1, \frac{\partial f}{\partial e}) \tag{4.40}$$

We then remark that this choice of coordinates is for convenience, and the same result will be obtained for an alternative choice. Thus, the result is without bias. Inheriting the inner product from $\mathcal{R}^6$

$$g_{11} = (\frac{\partial f}{\partial a}, \frac{\partial f}{\partial a}) = 1 + 0^2 + 0^2 + 0^2 + 0^2 + (\frac{\partial f}{\partial a})^2 \tag{4.41}$$
$$g_{12} = (\frac{\partial f}{\partial a}, \frac{\partial f}{\partial b}) = \frac{\partial f}{\partial a}\frac{\partial f}{\partial b} \tag{4.42}$$
$$g_{13} = (\frac{\partial f}{\partial a}, \frac{\partial f}{\partial c}) = \frac{\partial f}{\partial a}\frac{\partial f}{\partial c} \tag{4.43}$$
$$\vdots \tag{4.44}$$
$$g_{22} = (\frac{\partial f}{\partial b}, \frac{\partial f}{\partial b}) = 0^2 + 1 + 0^2 + 0^2 + 0^2 + (\frac{\partial f}{\partial b})^2 \tag{4.45}$$

where

$$G = \begin{pmatrix} g_{11} & \cdots & g_{15} \\ \vdots & \ddots & \vdots \\ g_{51} & \cdots & g_{55} \end{pmatrix}$$

Note:

$$\nabla h = (\frac{\partial P}{\partial a}, \frac{\partial P}{\partial b}, \ldots, \frac{\partial P}{\partial e}) \tag{4.46}$$

$$
\begin{aligned}
Slope\Big|_{Un-biased} &= \sqrt{\nabla h G^{-1} \nabla h^T} \\
&= \sqrt{\Big(\frac{\partial P}{\partial a} \cdots \quad \frac{\partial P}{\partial e}\Big) G^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \vdots \\ \frac{\partial P}{\partial e} \end{pmatrix}} \\
&= \sqrt{\begin{pmatrix} \frac{\partial P}{\partial a} \\ \vdots \\ \frac{\partial P}{\partial e} \end{pmatrix}^T G^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \vdots \\ \frac{\partial P}{\partial e} \end{pmatrix}} \\
&= \sqrt{\Big(\frac{\partial P}{\partial a} \cdots \quad \frac{\partial P}{\partial e}\Big) \begin{pmatrix} g_{11} & \cdots & g_{15} \\ \vdots & \ddots & \vdots \\ g_{51} & \cdots & g_{55} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial P}{\partial a} \\ \vdots \\ \frac{\partial P}{\partial e} \end{pmatrix}}
\end{aligned} \tag{4.47}
$$

D. Simulation Results

The results are obtained using different nominals, and different types of correlations (positive and negative). They are combined with different values of ϵ_N allowing us to generate eight examples of gradient distributions for the non-stationary cases (biased and unbiased approach superimposed), detailed in table VI.

In each case, the results are obtained for 100 bins, deploying a bin-size of $4 * 10^{-2}$ over range of 0-4 generates 100 bins. For each graph, we represent the normalized values of each sample densities. Table VII gives the corresponding means for both the biased and unbiased case, for each case scenario.

For each sample density, its arithmetic mean and harmonic mean are calculated. From table VII we see that, simulations in the biased domain expose the presence of finite but few outliers on points of the parameter surface (covariance matrix is positive definite), due to these the arithmetic mean of gradient is of the order of $10^5 - 10^6$,

Table VI. All the different scenarios for non-stationary

Case	a_0	b_0	c_0	d_0	e_0	f_0	ϵ_N
1	2	2	2	0.2	0.2	0.2	0.147773
2	2	2	2	0.2	0.2	-0.2	0.147773
3	2	2	2	-0.2	-0.2	0.2	0.147773
4	2	2	2	0.5	0.5	0.2	0.150312
5	2	2	2	0.2	0.2	0.5	0.149048
6	4	4	4	0.2	0.2	0.2	0.294447
7	2	2	2	0.2	0.2	0.7	0.150491
8	2	2	2	0.2	0.2	0.2	0.295546

Table VII. Means for each case scenario biased and unbiased approach

	Biased		Unbiased	
Case	Mean	Harmonic Mean	Mean	Harmonic Mean
1	$3.44917 * 10^5$	1.16537	0.830628	0.799697
2	$3.97407 * 10^5$	1.16919	0.830628	0.799697
3	$3.44917 * 10^5$	1.16537	0.830628	0.799697
4	$2.08469 * 10^5$	1.1385	0.827619	0.79686
5	$2.45522 * 10^6$	1.3037	0.839663	0.808334
6	$2.09488 * 10^5$	1.15966	0.831031	0.800077
7	$1.79842 * 10^6$	1.38615	0.839422	0.808101
8	$4.99711 * 10^5$	1.23752	0.840502	0.809145

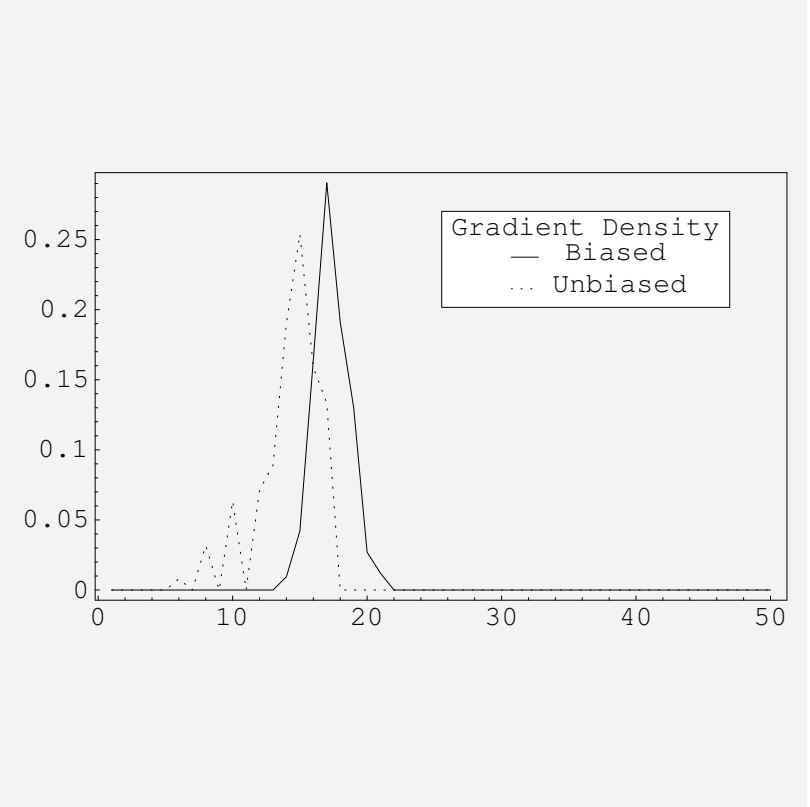

Fig. 48. Gradient densities for the non-stationary example: case 1

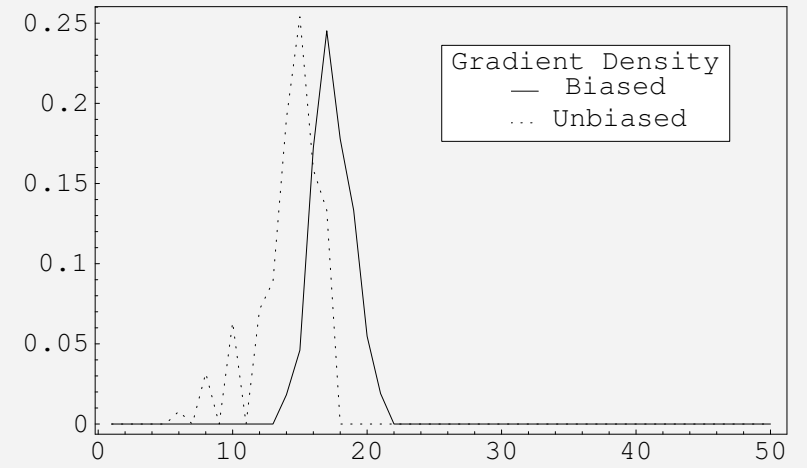

Fig. 49. Gradient densities for the non-stationary example: case 2

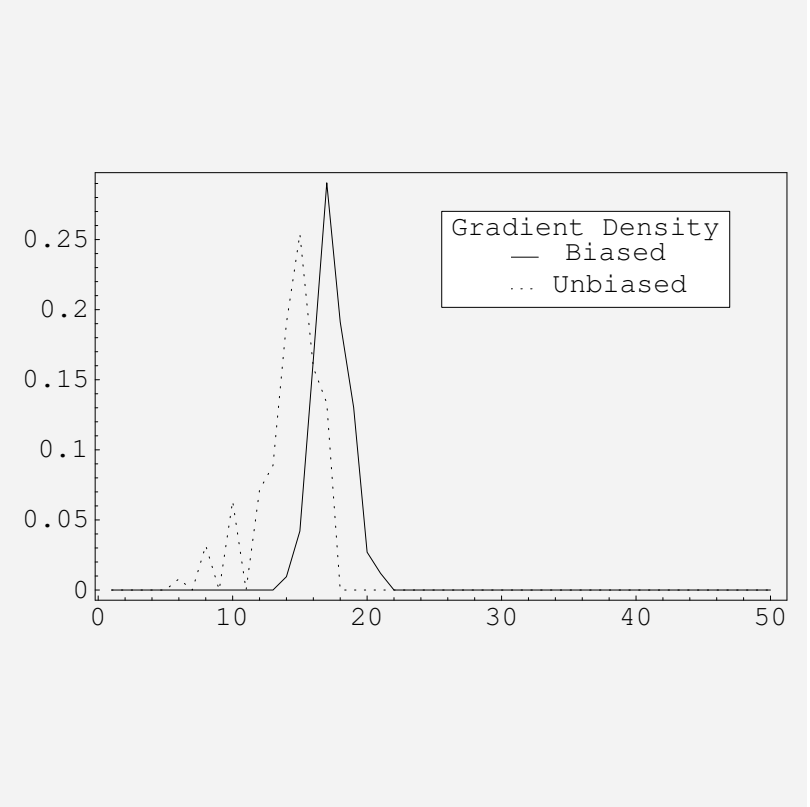

Fig. 50. Gradient densities for the non-stationary example: case 3

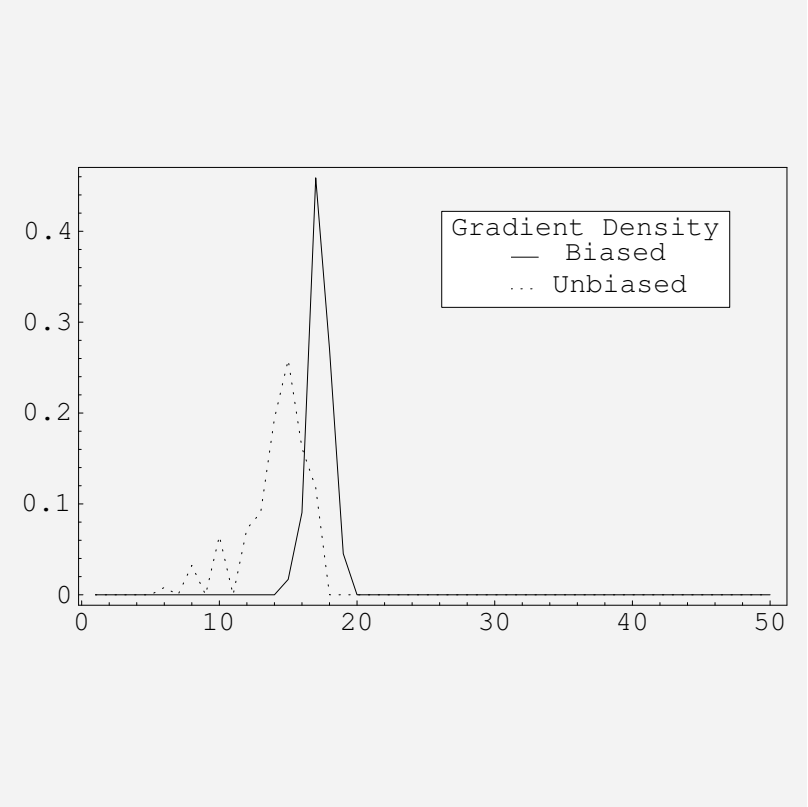

Fig. 51. Gradient densities for the non-stationary example: case 4

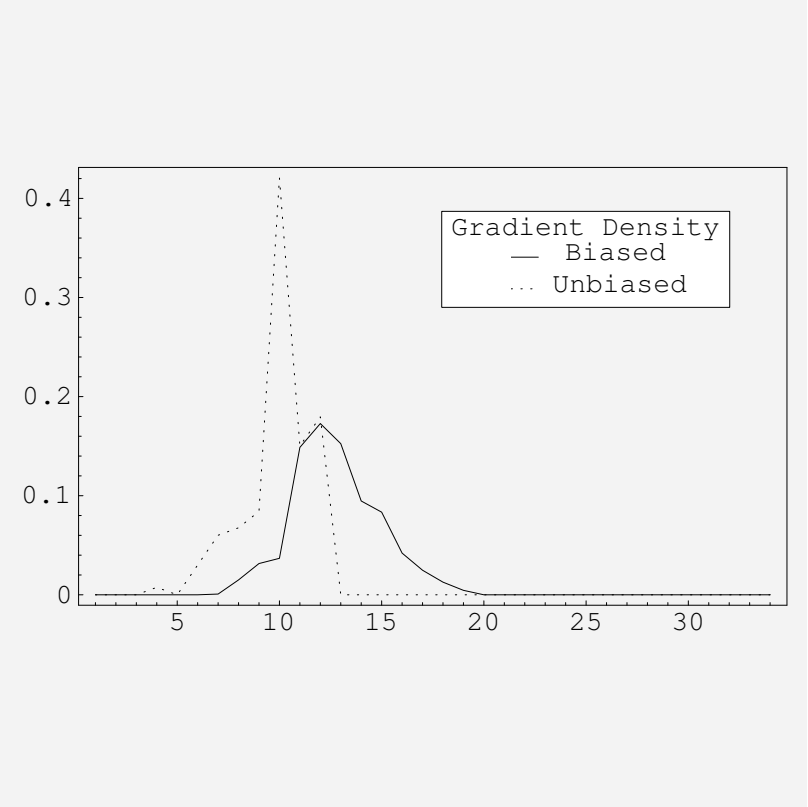

Fig. 52. Gradient densities for the non-stationary example: case 5

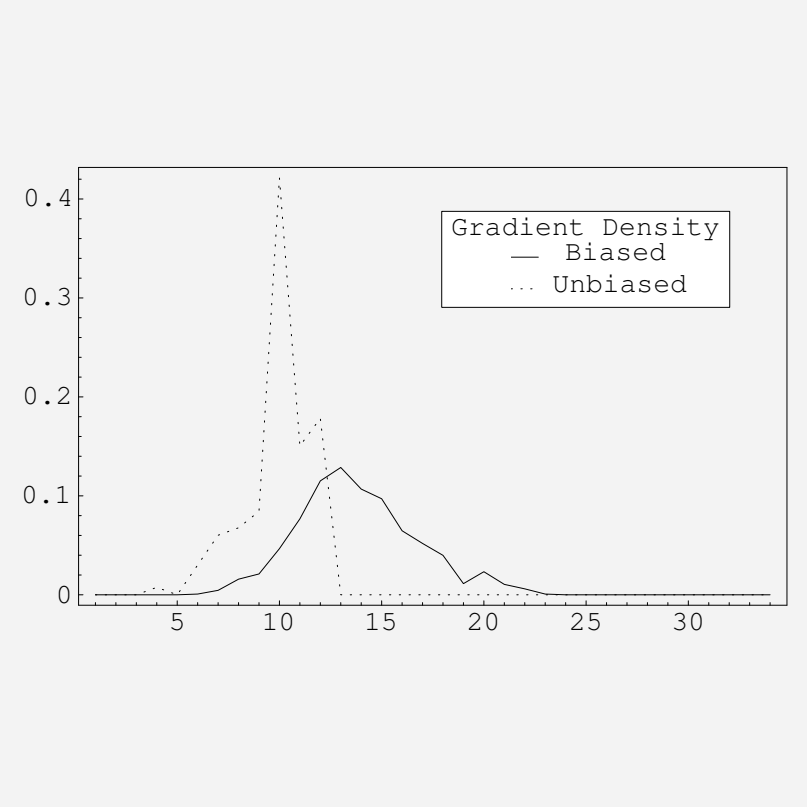

Fig. 53. Gradient densities for the non-stationary example: case 6

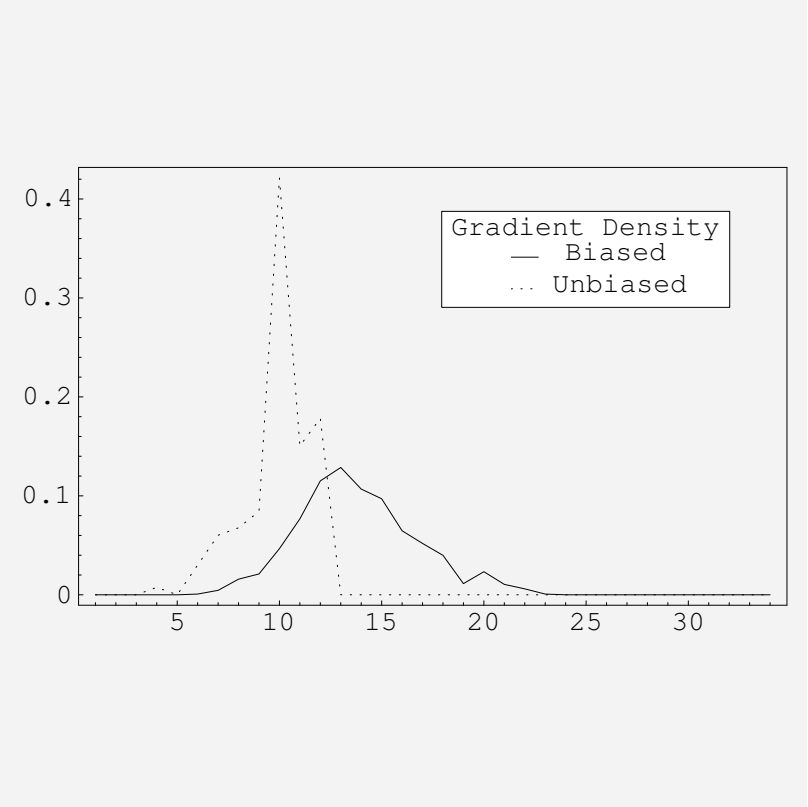

Fig. 54. Gradient densities for the non-stationary example: case 7

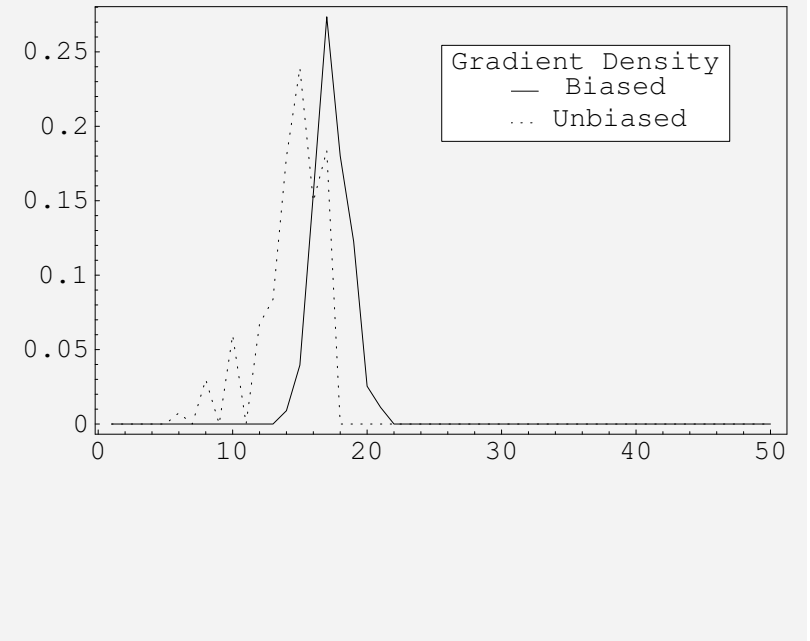

Fig. 55. Gradient densities for the non-stationary example: case 8

implying a highly un-robust system at those points.

In statistics and mathematics, the harmonic mean [63, 65, 64] (formerly sometimes called the sub-contrary mean) is one of several kinds of average. Typically, it is appropriate for situations when the average of rates is desired. The harmonic mean is the number of variables divided by the sum of the reciprocals of the variables. Since, we know that robustness (gradient) is a rate, we leverage this metric. In certain situations, the harmonic mean provides the truest average. Some important applications are average speed [66], in electrical theory: resistance of multiple-connected resistors in parallel, and in finance and investments: the harmonic mean is used to calculate the average cost of shares purchased over a period of time.

The harmonic mean H of the positive real numbers $x_1, x_2, ...$ is defined to be:

$$\frac{1}{H} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{x_i} \tag{4.48}$$

Since the harmonic mean of a list of numbers tends strongly toward the least elements of the list, it tends (compared to the arithmetic mean) to mitigate the impact of large outliers and aggravate the impact of small ones.

This is expected to allow us to state on the measure of robustness. All the results are presented in figures 48, 49, 50, 51, 52, 53, 54, 55. One of the requirements for all those sample densities is the need for positive definiteness for all sets of points. Each corresponding C matrix formed with those points has the particularity of being real and symmetrical forming an Hermitian matrix. The Hermitian matrix has special properties that could be used by one to verify that the sets of points are positive definite. One would only need to compute all the eigenvalues, knowing that if all the eigenvalues are positive, then the matrix is positive definite. Note that any not-positive definite sets of points are dropped by the simulation program.

E. Conclusion

We have seen that the Huber Strassen method resists admitting non-stationarity. In engineering systems there is an important class of applications where non stationarity can be admitted through the use of higher dimensionality. Covariance and correlation matrices arise quite often, and the differences between the stationary and non-stationary case is the dimension of the matrix. We have illustrated the admission of non stationarity for the linear predictor where the uncertain knowledge can be modeled as a manifold, nevertheless of higher dimension than for the stationary case. The chapter illustrates how, for example, gradient can be regarded as a random variable and a gradient probability density generated by means of a distribution, allowing conclusions regarding robustness to be drawn from statistical metrics such as arithmetic mean and harmonic mean, in addition to the form factor of the density functions.

This chapter also proves to go beyond expected results and exposes some remarkable facts. Simulations in the biased domain expose the presence of finite but few outliers on points of the parameter surface (covariance matrix is positive definite), where the gradient is of the order of $10^5 - 10^6$, implying a highly un-robust system. Moreover, no such phenomenon is seen over the same points in unbiased model. This however is not seen in [56]. This would imply that the bias in the system is highly pronounced for non-stationary model in our application. We can see that in the case of biased perturbations model, dramatic variations are translated to the Euclidean space. It is with these variations that the heavy bias enters the system, because the manifold has been severely altered. This requires that system engineers exclusively employ unbiased model, where some residual stationarity will be encountered.

However, as we have seen earlier, unbiased approach involves a matrix inversion.

The complexity of matrix inversion in hardware becomes prohibitive for real time applications and large values of matrix dimension, as we have encountered in the non-stationary model. This operation is inherently complex to design in hardware domain, more so in real time systems and requires fast, pipelined and scalable hardware architecture. Bridging the gap between these two highly specialized domains allows more degrees of freedom when determining the final solution. As remarked earlier, this development is beyond the scope of this dissertation.

We have also shown that, if one is constrained to use biased approach, designer can null off the out-lier points by employing harmonic means, (harmonic mean filters the impact of large outliers).

The next step of this dissertation establishes a crucial nexus to our prior research: we address the fundamental question, "Is the stationary case more robust than the non-stationary?"

CHAPTER V

STATIONARITY VERSUS NON-STATIONARITY: TOO ROBUST OR NOT TOO ROBUST?

Classically, the noise models were stationary Gaussian with a parametric assumption (all necessary parameters assumed known), but it was quickly realized that some relaxation of such assumptions was necessary. For example, while a Gaussian model might be useful as a first step, it is certainly desirable to allow the entries in its covariance matrix to be imperfectly known and to admit residual non-stationarity. We have seen that this extension is not trivial, nevertheless, the real work involves in what is done to compare the stationary and the non stationary, and, to determine if there are any conclusions (at least for specific examples) that can be drawn regarding the effect of non stationarity on robustness.

A. Recap of Mathematical Model for Stationary and Non-Stationary Systems

1. The Linear Prediction Filter

The linear prediction of a random variable $X[3] \equiv X_3$ in terms of $X[2] \equiv X_2$ and $X[1] \equiv X_1$ is:

$$\hat{X}[3] = k_1 X[1] + k_2 X[2] \tag{5.1}$$

where X[.] are Gaussian distributed random variables with covariance matrix C:

$$C = \begin{pmatrix} E[X_1^2] & E[X_1X_2] & E[X_1X_3] \\ E[X_1X_2] & E[X_2^2] & E[X_2X_3] \\ E[X_1X_3] & E[X_2X_3] & E[X_3^2] \end{pmatrix}$$

Applying Projection Theorem or Yule-Walker Equations, we obtain k_1 and k_2, and these design parameters/filter taps/coefficients are computed using nominal value of covariance matrix. The MSE is defined as

$$\begin{aligned} MSE &= E[[\hat{X}[3] - X[3]]^2] \\ &= E[[k_1 X[1] + k_2 X[2] - X[3]]^2] \end{aligned} \tag{5.2}$$

After few computations, the MSE statistic can easily be transformed into:

$$\begin{aligned} MSE &= E[X_3]^2 + k_1^2 E[X_1]^2 + k_2^2 E[X_2]^2 \\ &- 2k_1 E[X_1 X_3] - 2k_2 E[X_2 X_3] + 2k_1 k_2 E[X_1 X_2] \end{aligned} \tag{5.3}$$

2. The Stationary Case

For a Wide Sense Stationary Process, the correlation matrix is

$$C_S = \begin{pmatrix} A & B/\sqrt{2} & C/\sqrt{2} \\ B/\sqrt{2} & A & B/\sqrt{2} \\ C/\sqrt{2} & B/\sqrt{2} & A \end{pmatrix}$$

For the computation of the value of the design parameters k_1, k_2, one will employ nominal values, chosen according to the table VIII.

The parameter surface obtained using the Frobenius norm, for the stationary case, would be 2-ellipsoid in 3D:

$$\|E_S\|^2 = 3(A - A_0)^2 + 2(B - B_0)^2 + (C - C_0)^2 = {\epsilon_S}^2 \tag{5.4}$$

Table VIII. Nominal values for the stationary case

Correlation	Nominal Value
$E[X_1^2]$	A_0
$E[X_2^2]$	A_0
$E[X_3^2]$	A_0
$E[X_1X_2]$	$B_0/\sqrt{2}$
$E[X_2X_3]$	$B_0/\sqrt{2}$
$E[X_1X_3]$	$C_0/\sqrt{2}$

The MSE is:

$$\begin{aligned} MSE\Big|_{WSS} &= E[[\hat{X}[3] - X[3]]^2] \\ &= A[1 + k_1^2 + k_2^2] + \sqrt{2}Bk_2[k_1 - 1] - \sqrt{2}Ck_1 \\ &= P(A, B, C) \end{aligned} \tag{5.5}$$

3. The Non-Stationary Case

For non-stationary distribution correlation matrix, becomes:

$$C_N = \begin{pmatrix} E[X_1^2] & E[X_1X_2] & E[X_1X_3] \\ E[X_1X_2] & E[X_2^2] & E[X_2X_3] \\ E[X_1X_3] & E[X_2X_3] & E[X_3^2] \end{pmatrix}$$

where $X_1 \sim \mathcal{N}(0, a)$, $X_2 \sim \mathcal{N}(0, b)$ and $X_3 \sim \mathcal{N}(0, c)$. Thus,

$$C_N = \begin{pmatrix} a & d/\sqrt{2} & f/\sqrt{2} \\ d/\sqrt{2} & b & e/\sqrt{2} \\ f/\sqrt{2} & e/\sqrt{2} & c \end{pmatrix}$$

and $E[X_1X_2] = d/\sqrt{2}$, $E[X_1X_3] = f/\sqrt{2}$, $E[X_2X_3] = f/\sqrt{2}$. For the computation of the value of the design parameters k_1, k_2, one will employ nominal values, chosen according to the table IX.

The parameter surface obtained using the Frobenius norm, for the non-stationary

Table IX. Nominal values for the non-stationary case

Correlation	Nominal Value
$E[X_1^2]$	a_0
$E[X_2^2]$	b_0
$E[X_3^2]$	c_0
$E[X_1X_2]$	$d_0/\sqrt{2}$
$E[X_2X_3]$	$e_0/\sqrt{2}$
$E[X_1X_3]$	$f_0/\sqrt{2}$

case, would be a 5-sphere in 6D:

$$\|E_N\|^2 = (a-a_0)^2+(b-b_0)^2+(c-c_0)^2+(d-d_0)^2+(e-e_0)^2+(f-f_0)^2 = {\epsilon_N}^2 \quad (5.6)$$

Thus,

$$\begin{aligned} f &= f_0 \pm \sqrt{{\epsilon_N}^2 - (a-a_0)^2 + (b-b_0)^2 + (c-c_0)^2 + (d-d_0)^2 + (e-e_0)^2} \\ f(a,b,c,d,e) &= f_0 \pm \sqrt{{\epsilon_N}^2 - (a-a_0)^2 + (b-b_0)^2 + (c-c_0)^2 + (d-d_0)^2 + (e-e_0)^2} \end{aligned}$$

For the non-stationary case MSE is equal to:

$$\begin{aligned} MSE &= E[X_3]^2 + k_1^2 E[X_1]^2 + k_2^2 E[X_2]^2 \\ &- 2k_1 E[X_1 X_3] - 2k_2 E[X_2 X_3] + 2k_1 k_2 E[X_1 X_2] \\ MSE\Big|_{NS} &= c + k_1^2 a + k_2^2 b \\ &- 2k_1 f/\sqrt{2} - 2k_2 e/\sqrt{2} + 2k_1 k_2 d/\sqrt{2} \\ MSE\Big|_{NS} &= c + k_1^2 a + k_2^2 b \\ &- 2k_1 f(a,b,c,d,e)/\sqrt{2} - 2k_2 e/\sqrt{2} + 2k_1 k_2 d/\sqrt{2} \end{aligned} \tag{5.7}$$

B. Matching ϵ_S and ϵ_N

From equations 5.4 and 5.6 we see that a major problem involves comparing the non stationarity robustness measure to the stationarity. Because of the difference in dimensionality, it would not be a good idea to employ ϵ - balls of the same radius when modeling; otherwise a bias may creep in. The size of the ball for the stationary and non-stationary cases must be matched so that we avoid an apple and orange comparison. For example, a ball of small radius is much more variable than one of large radius, in the sense that a small deviation from one point will lead to a larger gradient. Thus, curvature becomes relevant, and one can integrate Gaussian curvatures over the manifold to obtain total curvature. If one then divides by the volume of the manifold, average curvature is obtained. For the specific example of the linear predictor, we propose choosing the radius ϵ_S of the stationary model and the

radius ϵ_N of the higher order non-stationary model so that equal average curvatures are obtained. Only then will the resultant robustness measures admit comparison.

For two-dimensional surfaces embedded in R^3, there are two kinds of curvature: Gaussian curvature and Mean curvature. To compute these at a given point of the surface, one needs to consider the intersection of the surface with a plane containing a fixed normal vector at the point. This intersection is a plane curve and has a curvature. If we vary the plane, this curvature will change. Furthermore, there are two extremal values - the maximal and the minimal curvature, called the principal curvatures, $k1$ and $k2$, the extremal directions are called principal directions. Here we adopt the convention that a curvature is taken to be positive if the curve turns in the same direction as the surface's chosen normal, otherwise negative.

The Gaussian curvature [67], named after Carl Friedrich Gauss, is equal to the product of the principal curvatures, $k1 * k2$. It is positive for spheres, negative for one sheet hyperboloids, and zero for planes. It determines whether a surface is locally convex (when it is positive) or locally saddle (when it is negative).

The above definition of Gaussian curvature is extrinsic in that it uses the surface's embedding in R^3, normal vectors, external planes etc. Gaussian curvature is, however, in fact an intrinsic property of the surface. This means it does not depend on the particular embedding of the surface. Intuitively, this means that ants living on the surface could determine the Gaussian curvature. Formally, Gaussian curvature only depends on the Riemannian metric of the surface (see [68]).

The matching of the values of ϵ is necessary to enable one to compare the obtained results. This can be done by looking at the average Gauss-Kronecker curvature over a surface. The curvature at a point on a manifold is given by:

$$K_n = \lambda_1 ... \lambda_n \tag{5.8}$$

where n is the dimension of the manifold and the λ_i are the *principle curvatures*. The average curvature is then equal to:

$$K_{n,Av} = \frac{\int_M K_n dV_n}{volume\ M} \tag{5.9}$$

where the volume is actually the "surface area" of the surface parameter. Since the curvature is an n-dimensional concept ($K_n = \prod_{i=1}^{n} \lambda_i$), we can equate:

$$(K_{5,Av})^{1/5} = (K_{2,Av})^{1/2} \tag{5.10}$$

Since n=5 has $m_5 = sphere\ of\ radius, \epsilon_N$, we have $K_n = (\frac{1}{\epsilon_N})^5$, so:

$$\begin{aligned} K_{5,Av} &= \frac{\int_{m_5} \epsilon_N^{-5} dV_5}{volume\ m_5} \\ &= \frac{\epsilon_N^{-5} \int_{m_5} dV_5}{volume\ m_5} \\ &= \frac{\epsilon_N^{-5}\ volume\ m_5}{volume\ m_5} \\ &= \epsilon_N^{-5} \end{aligned} \tag{5.11}$$

For the two dimensions, we have to use the Gauss-Bonnet theorem [69, 70, 71] because the manifold is non spherical. The Gauss-Bonnet theorem in differential geometry is an important statement about surfaces which connects their geometry (in the sense of curvature) to their topology (in the sense of the Euler characteristic).

Suppose M is a compact two-dimensional orientable Riemannian manifold with boundary ∂M. Denote by K the Gaussian curvature at points of M, and by k_g the geodesic curvature at points of ∂M. Then, the Gauss-Bonnet yields:

$$\int_M K dA + \int_{\partial M} k_g dS = 2\pi\chi(M) \tag{5.12}$$

where $\chi(M)$ is the Euler characteristic of M.

The theorem applies in particular if the manifold does not have a boundary, in

which case the integral $\int_{\partial M} k_g \, ds$ can be omitted. Therefore, for two dimensions:

$$\chi(M) = \frac{1}{2\pi} \int_{M_2} K_2 dV_2 \tag{5.13}$$

Since M_2 (2 dimension ellipsoid) is diffeomorphic to a 2-sphere, $\chi(M_2) = 2$, thus:

$$2 = \frac{1}{2\pi} \int_{M_2} K_2 dV_2 \tag{5.14}$$

or

$$4\pi = \int_{M_2} K_2 dV_2 \tag{5.15}$$

Also, we know that:

$$\begin{aligned} K_{2,Av} &= \frac{\int_{M_2} K_2 dV_2}{vol\ M_2} \\ &= \frac{4\pi}{vol\ M_2} \\ &= \frac{4\pi}{surface\ area\ of\ (3A^2 + 2B^2 + C^2 = \epsilon_S^2)} \\ &= \frac{4\pi}{Area(\epsilon_S)} \end{aligned} \tag{5.16}$$

Finally we obtain:

$$(\epsilon_N^{-5})^{1/5} = (\frac{4\pi}{Area(\epsilon_S)})^{1/2} \tag{5.17}$$

or

$$\epsilon_N = \sqrt{\frac{Area(\epsilon_S)}{4\pi}} \tag{5.18}$$

where $Area(\epsilon_S)$ is the surface area of $3A^2 + Bd^2 + C^2 = \epsilon_S^2$.

C. Surface Area of Ellipsoid

The general ellipsoid [72], also called a triaxial ellipsoid, is a quadratic surface which is given in Cartesian coordinates by

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{5.19}$$

where the semi-axes are of lengths a, b, and $c, (a > b > c)$. The surface area of an ellipsoid is given by

$$A = 2\pi c^2 + \frac{2\pi b}{\sqrt{a^2 - c^2}}[(a^2 - c^2)E(am(\theta), k) + c^2\theta] \tag{5.20}$$

where $E(\phi, k)$ is a incomplete elliptic integral of the second kind, $am(\phi)$ is the Jacobi amplitude with modulus k,

$$\begin{aligned} e_1^2 &\equiv \frac{a^2 - c^2}{a^2} \\ e_2^2 &\equiv \frac{b^2 - c^2}{b^2} \\ k &\equiv \frac{e_2}{e_1} \end{aligned}$$

θ is given by

$$e_1 = sn(\theta, k) \tag{5.21}$$

where $sn(\theta, k)$ is a Jacobi Elliptic function. Applying the above to the ellipsoid for Stationary case :

$$\begin{aligned} 3A^2 + 2B^2 + C^2 &= \epsilon_S^2 \\ \frac{A^2}{(\epsilon_S/\sqrt{3})^2} + \frac{B^2}{(\epsilon_S/\sqrt{2})^2} + \frac{C^2}{\epsilon_S^2} &= 1 \end{aligned} \tag{5.22}$$

Here, $a = \epsilon_S, b = \epsilon_S/\sqrt{2}, c = \epsilon_S/\sqrt{3}$.

Finally, we get:

$$\epsilon_N = 0.81 * \epsilon_S \tag{5.23}$$

Philosophically, this would mean, that, due to increased dimensionality in non-stationary-model, the radius of the sphere would be no more than 81% of the stationary case, in order for one to draw meaningful comparison.

D. Simulation Results

The results are obtained using different nominals (inheriting from previous examples)(see table X), and different types of correlations (positive and negative). They are combined with two different values of $\epsilon's$ (one ϵ_S and a smaller value of ϵ_N, matched to the ϵ_S) allowing us to generate eight examples of gradient distributions for the stationary and non-stationary cases superimposed.

In each case, the results are obtained using a bin-size such that size of $4 * 10^{-2}$, over range of 0-4, generates 100 bins. For each graph, we represent the normalized values of each sample densities because of the scaling effect on the results that ϵ_N (non-stationary approach) has.

For each Gradient density, its arithmetic mean is calculated (see table XI). This is expected to allow us to state on the measure of robustness. All the results are presented in figures 56, 57, 58, 59, 60, 61, 62 and 63. We have also illustrated the comparison for biased case, in figures 64, 65, 66, 67, 68, 69, 70 and 71, however for analysis we exclusively use unbiased model. One of the requirements for all those sample densities is the need for positive definiteness for all sets of points. Each corresponding C matrix formed with those points has the particularity of being real and symmetrical forming an Hermitian matrix. The Hermitian matrix has special properties that could be used by one to verify that the sets of points are positive

Table X. All the different case scenarios for both stationary and non-stationary

Case	a_0	b_0	c_0	d_0	e_0	f_0	ϵ_S	ϵ_N
1	2	2	2	0.2	0.2	0.2	0.174069	0.147773
2	2	2	2	0.2	0.2	-0.2	0.174069	0.147773
3	2	2	2	-0.2	-0.2	0.2	0.174069	0.147773
4	2	2	2	0.5	0.5	0.2	0.177059	0.150312
5	2	2	2	0.2	0.2	0.5	0.175570	0.149048
6	4	4	4	0.2	0.2	0.2	0.346843	0.294447
7	2	2	2	0.2	0.2	0.7	0.177271	0.150491
8	2	2	2	0.2	0.2	0.2	0.348138	0.295546

Table XI. Means for each case scenario for both the stationary and non-stationary approach

Case	Mean	
	Stationary	Non-Stationary
1	0.783702	0.830628
2	0.789037	0.830628
3	0.783448	0.830628
4	0.81791	0.827619
5	0.816654	0.839663
6	0.774322	0.831031
7	0.856068	0.839422
8	0.783702	0.840502

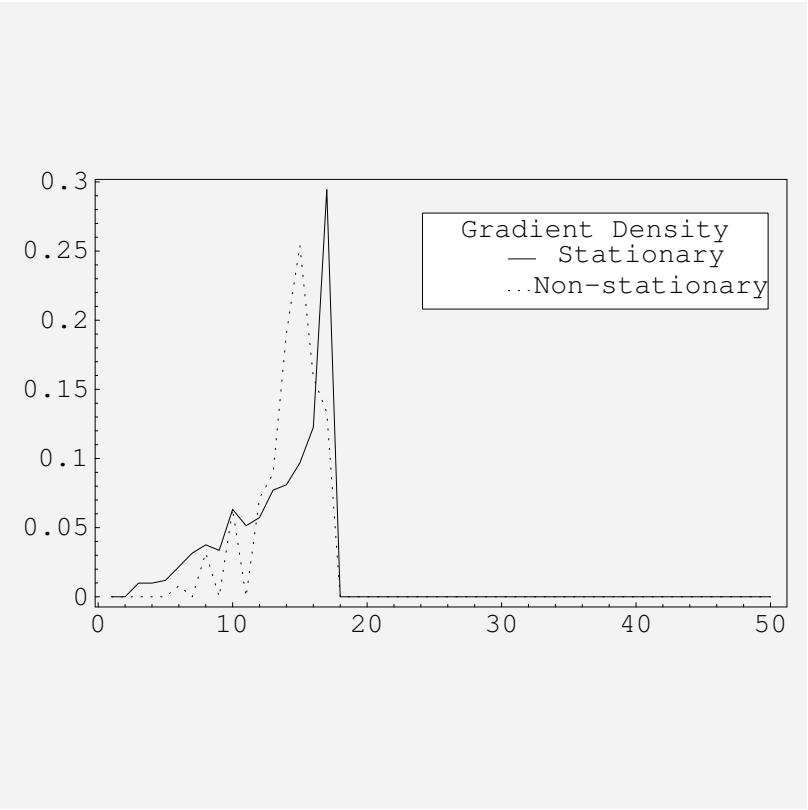

Fig. 56. Gradient densities for stationary vs. non-stationary (unbiased): case 1

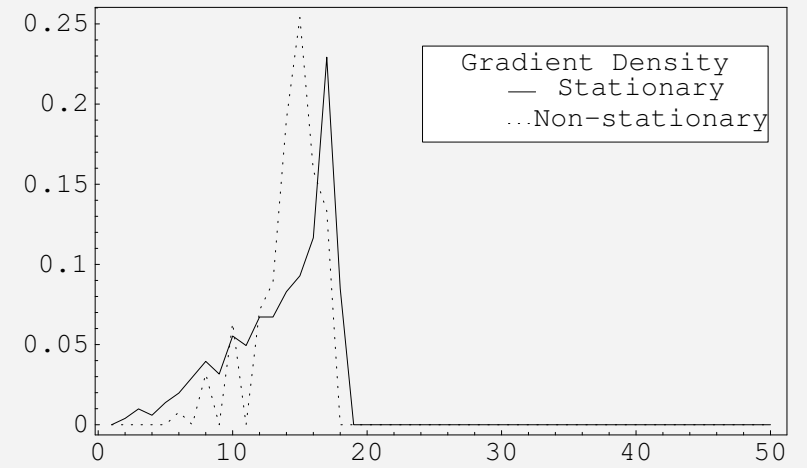

Fig. 57. Gradient densities for stationary vs. non-stationary (unbiased): case 2

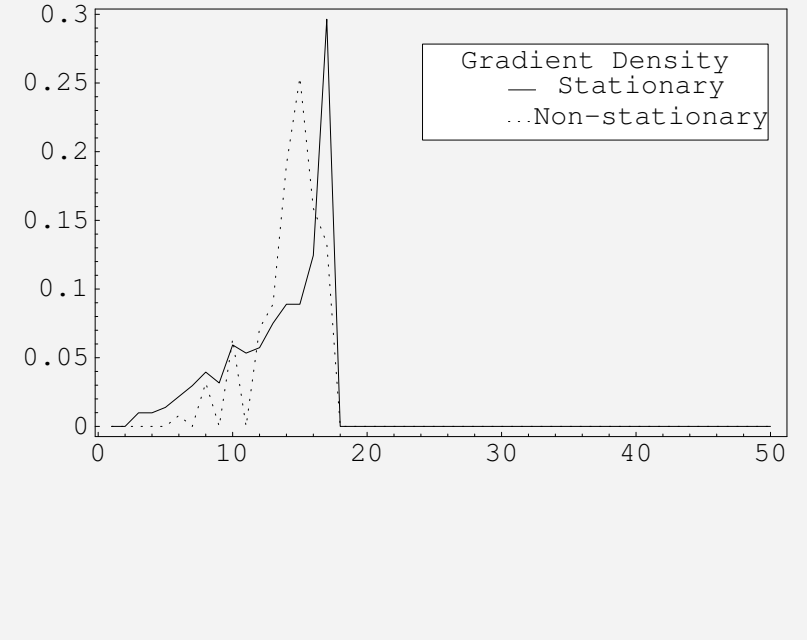

Fig. 58. Gradient densities for stationary vs. non-stationary (unbiased): case 3

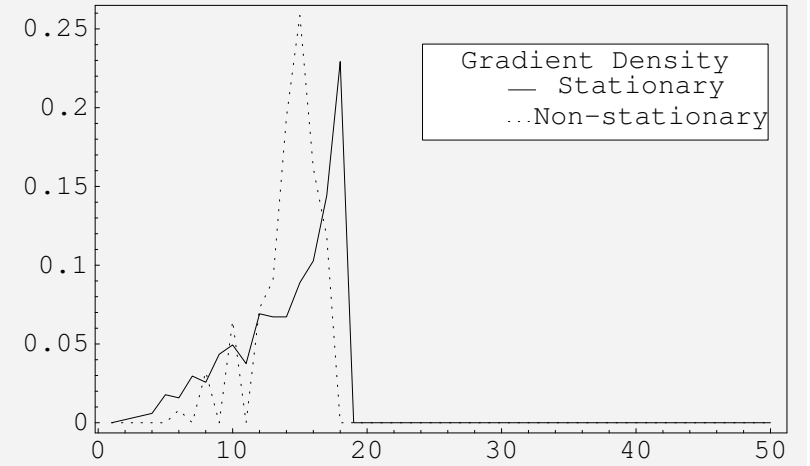

Fig. 59. Gradient densities for stationary vs. non-stationary (unbiased): case 4

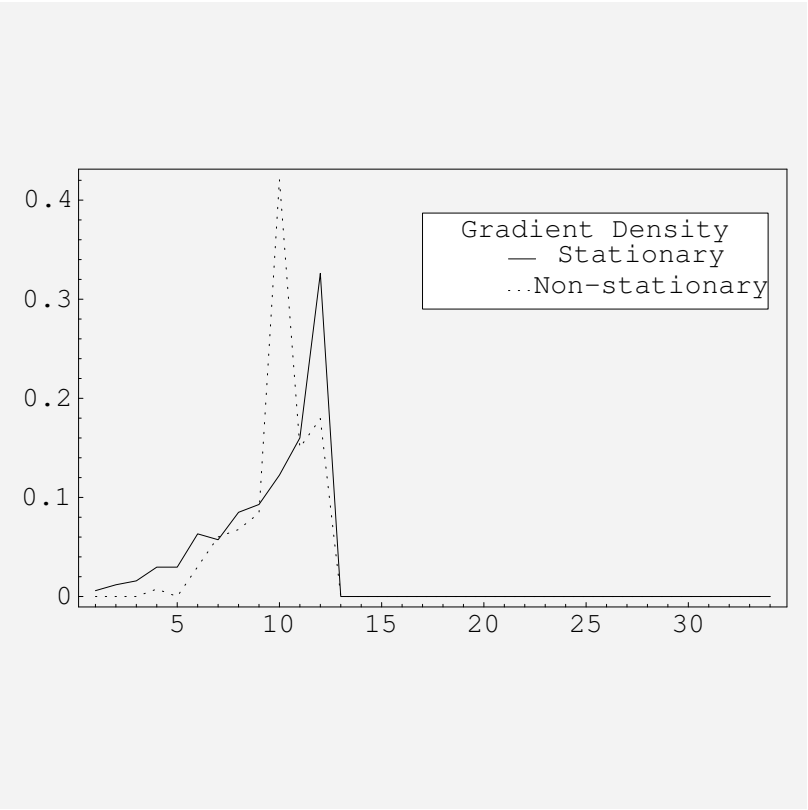

Fig. 60. Gradient densities for stationary vs. non-stationary (unbiased): case 5

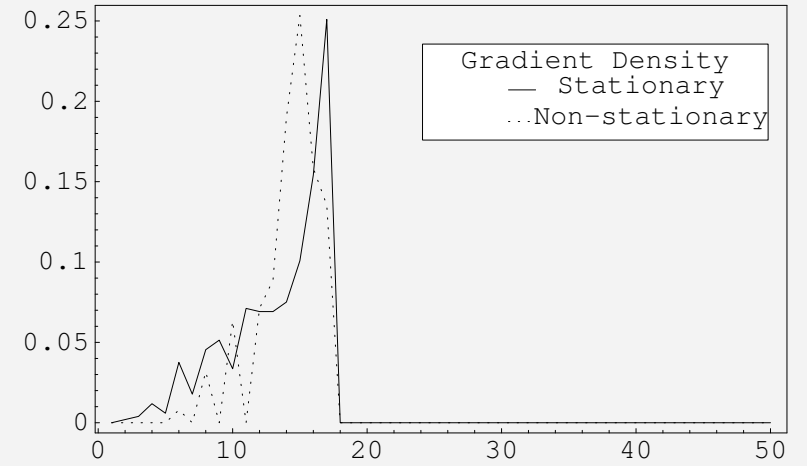

Fig. 61. Gradient densities for stationary vs. non-stationary (unbiased): case 6

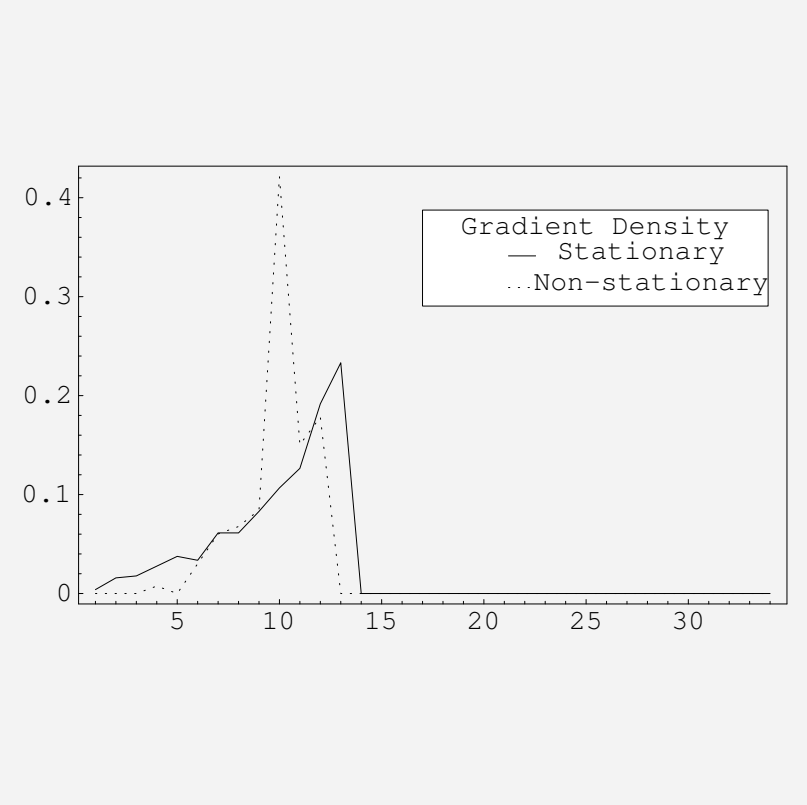

Fig. 62. Gradient densities for stationary vs. non-stationary (unbiased): case 7

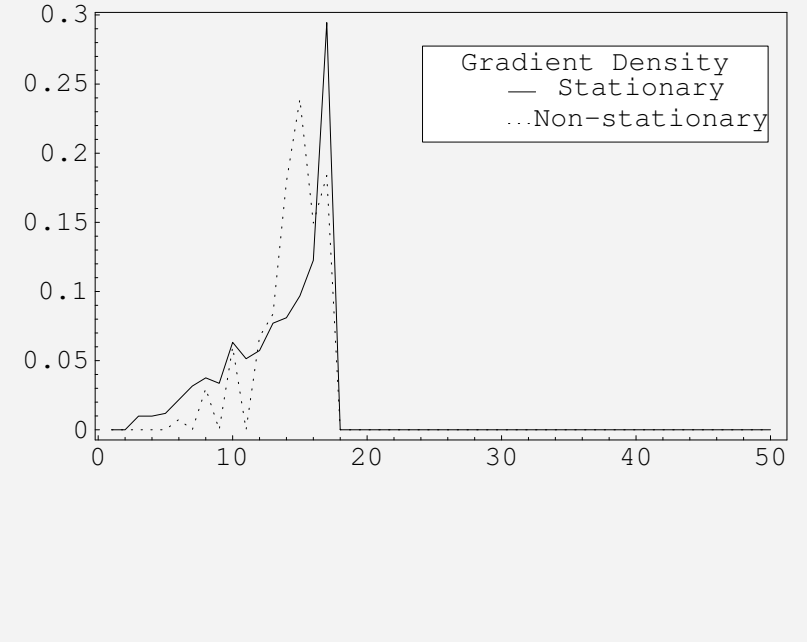

Fig. 63. Gradient densities for stationary vs. non-stationary (unbiased): case 8

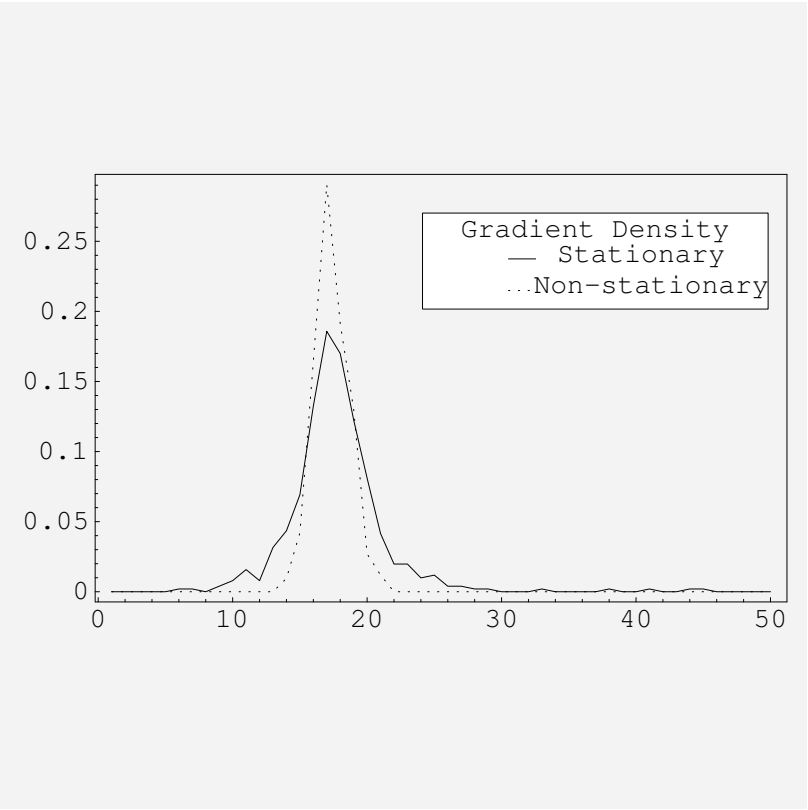

Fig. 64. Gradient densities for stationary vs. non-stationary (biased): case 1

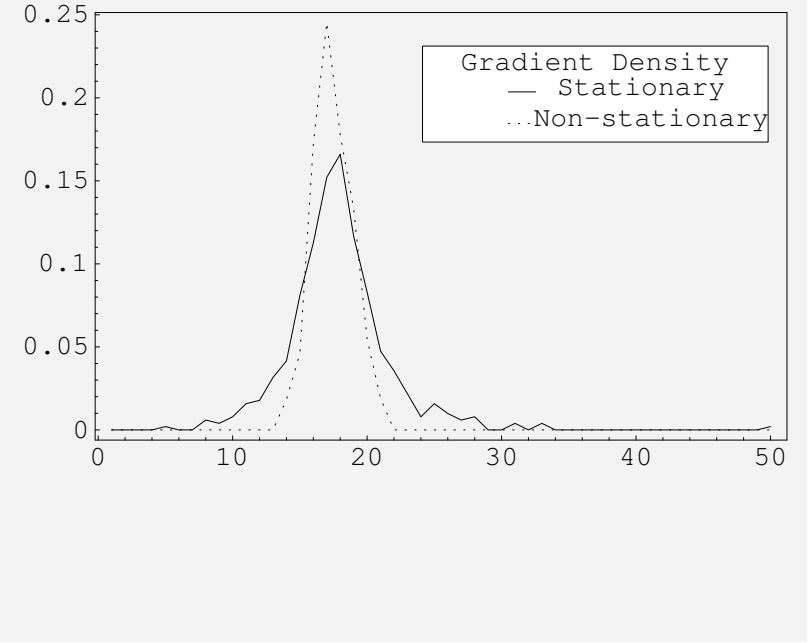

Fig. 65. Gradient densities for stationary vs. non-stationary (biased): case 2

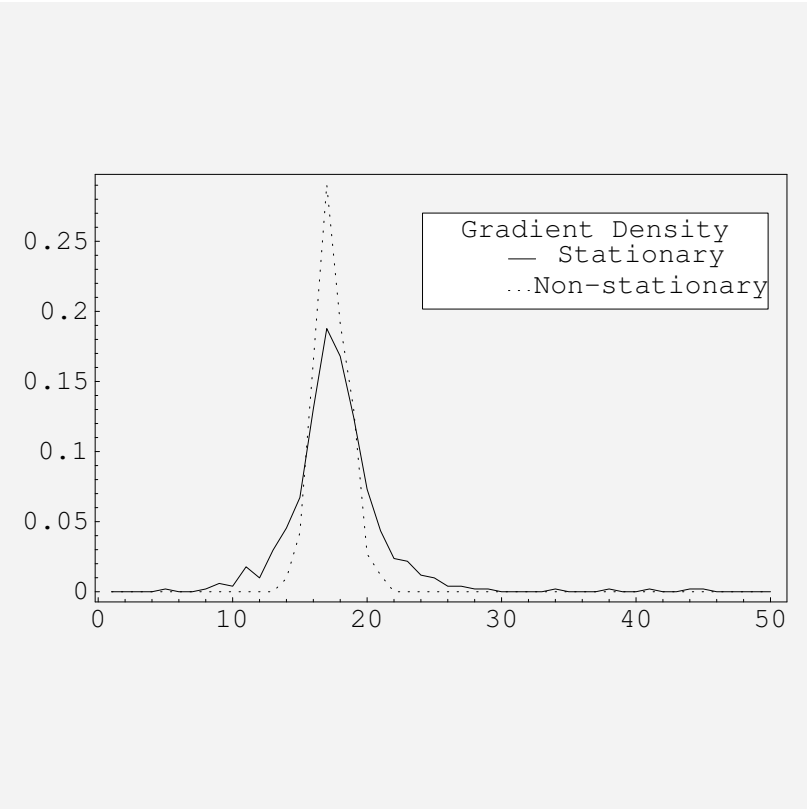

Fig. 66. Gradient densities for stationary vs. non-stationary (biased): case 3

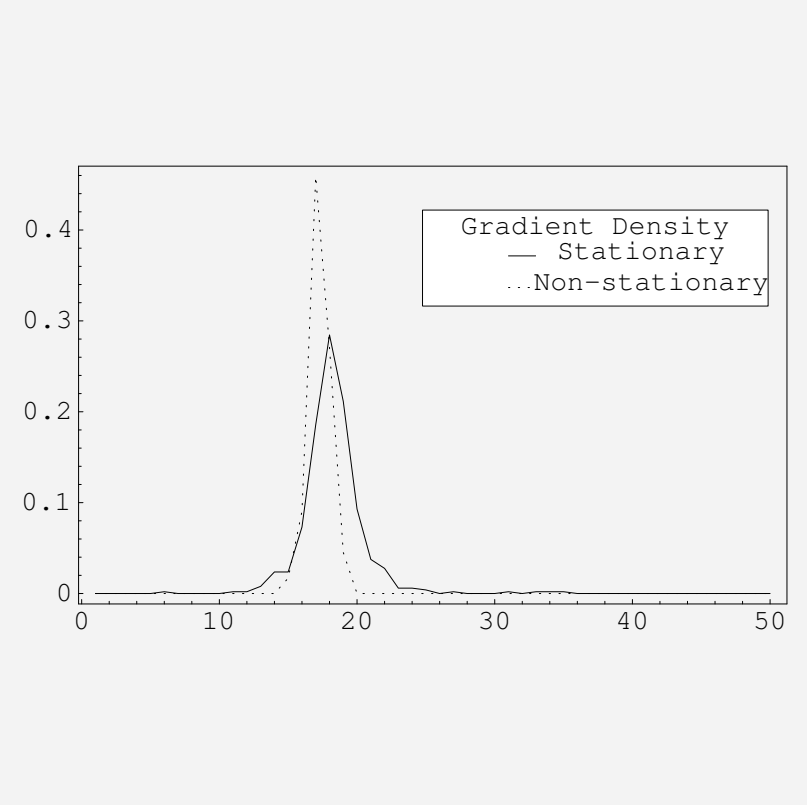

Fig. 67. Gradient densities for stationary vs. non-stationary (biased): case 4

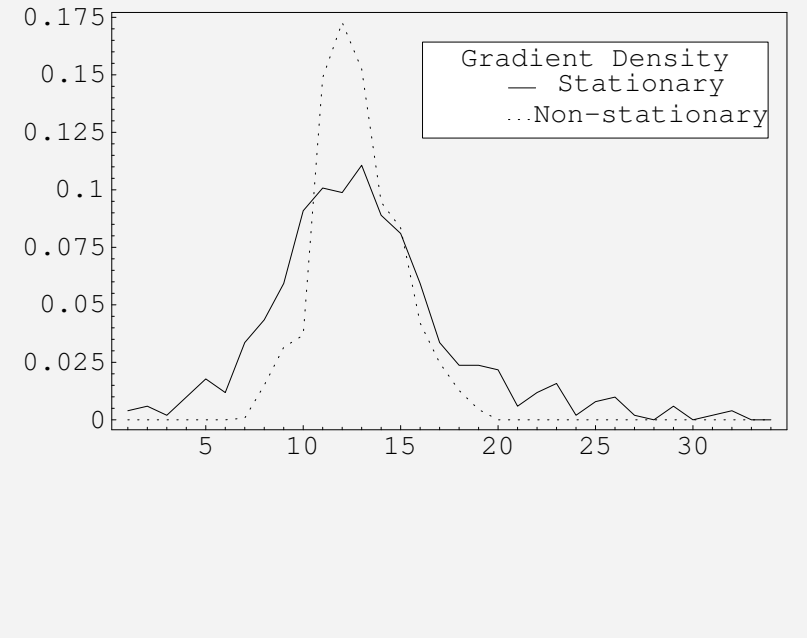

Fig. 68. Gradient densities for stationary vs. non-stationary (biased): case 5

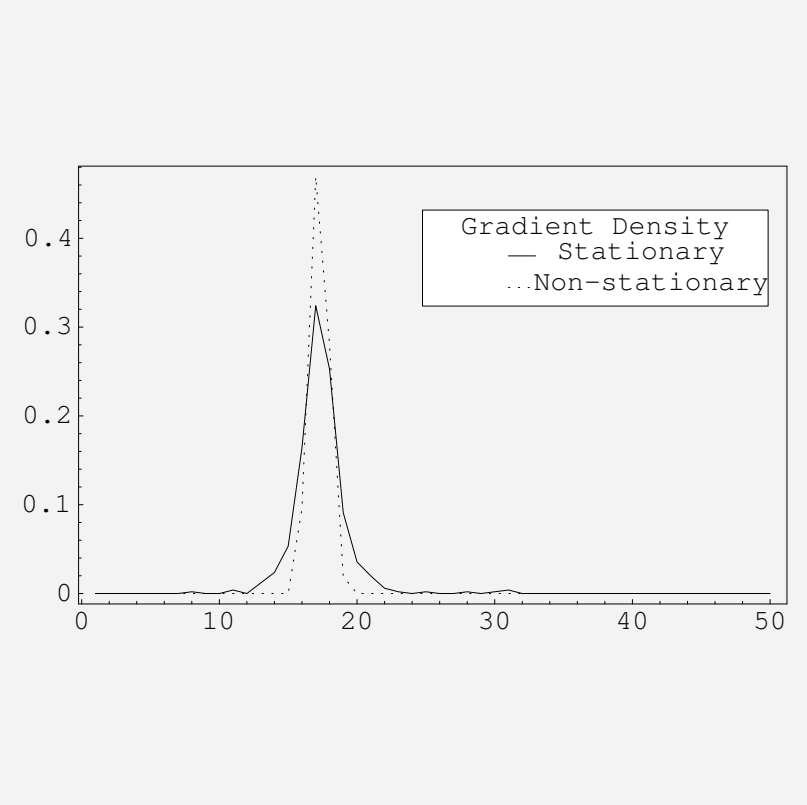

Fig. 69. Gradient densities for stationary vs. non-stationary (biased): case 6

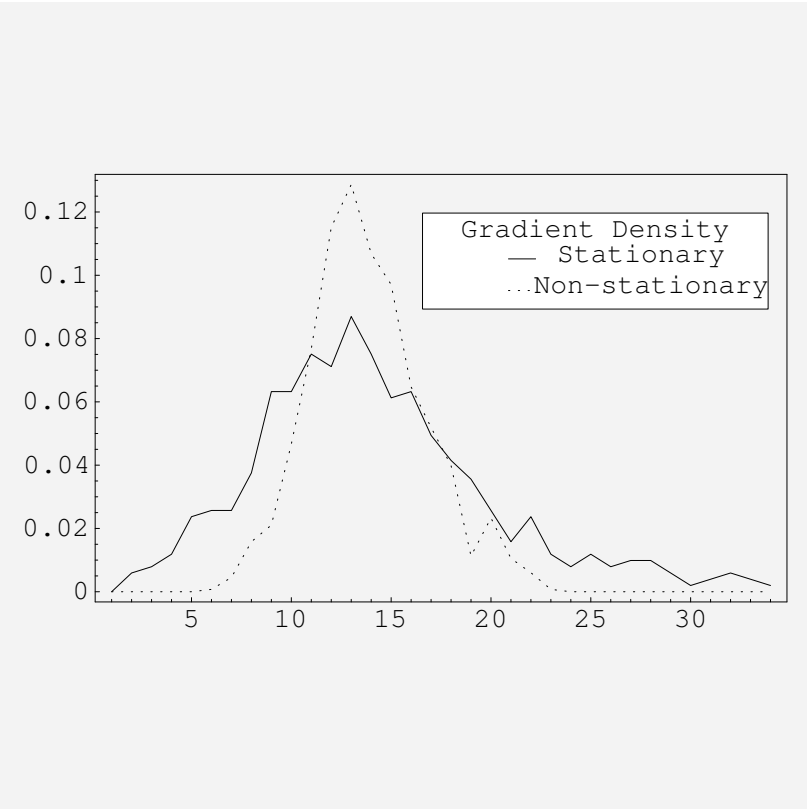

Fig. 70. Gradient densities for stationary vs. non-stationary (biased): case 7

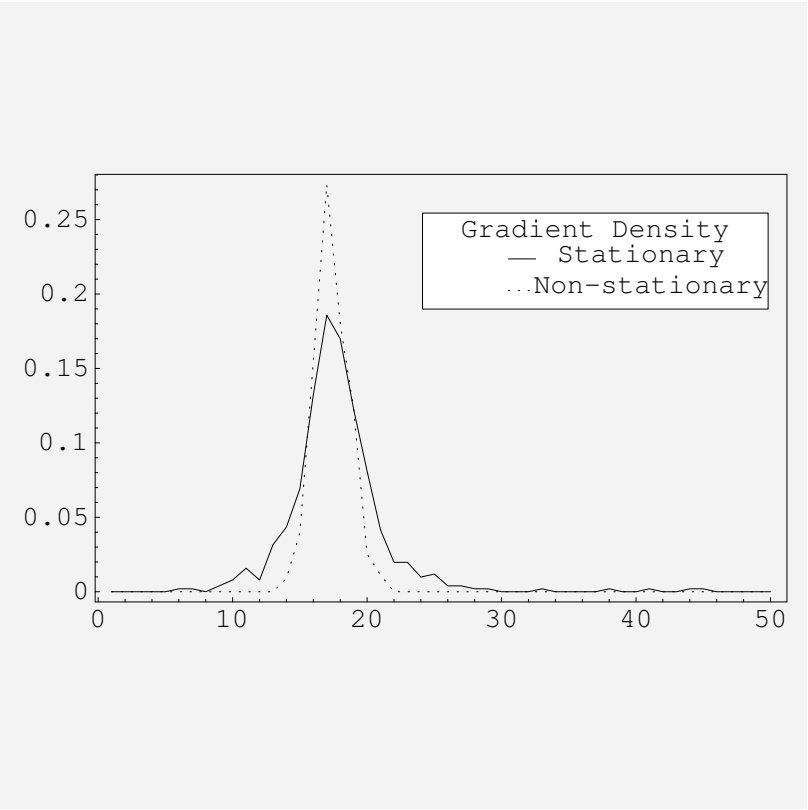

Fig. 71. Gradient densities for stationary vs. non-stationary (biased): case 8

definite. One would only need to compute all the eigenvalues, knowing that if all the eigenvalues are positive, then the matrix is positive definite. Note that any not-positive definite sets of points are automatically dropped by the simulation.

1. Median, Harmonic Mean and Confidence Bounds

While the form factor of density curves can be intriguing, it is not readily apparent how best to utilize them so as to address the fundamental question, "Is the stationary case more robust than the non-stationary?" This situation is complicated by the differing dimensionality that leads to the creation of the distributions. The introduction of the notions of median, mode and confidence bound will also aid in the making of the decision on the robustness of the algorithm scheme.

In probability theory and statistics, the median is a number that separates the highest half of a sample, a population, or a probability distribution from the lowest half. More precisely half of the population will have values less than or equal to the median and half of the population will have values equal to or greater than the median [73].

In statistics and mathematics, the harmonic mean [63, 65, 64] (formerly sometimes called the sub-contrary mean) is one of several kinds of average. Typically, it is appropriate for situations when the average of rates is desired. The harmonic mean is the number of variables divided by the sum of the reciprocals of the variables. Since, we know that robustness (gradient) is a rate, we leverage this metric. In certain situations, the harmonic mean provides the truest average. Some important applications are average speed [66], in electrical theory: resistance of multiple-connected resistors in parallel, and in finance and investments: the harmonic mean is used to calculate the average cost of shares purchased over a period of time. Since the harmonic mean of a list of numbers tends strongly toward the least elements of the list, it tends (com-

pared to the arithmetic mean) to mitigate the impact of large outliers and aggravate the impact of small ones.

A confidence bound is, for example, if X is a 95 percent upper one-sided bound, this would imply that ninety five percent of the population is less than X. If X is a ninety five percent lower one-sided bound, this would indicate that ninety percent of the population is greater than X. For the following results, we used a ninety five percent confidence bound that is an upper one sided bound (ninety five percent of the area under the curve that is represented by the sample density is on the left of that limit) [74].

The median, mode, and ninety five percent confidence bound are recalculated for all the previous simulations. A summary of the results are presented in tables XII and XIII.

2. Upper Bound on Change in MSE

While there are admittedly many ways one might use the distributions to evoke an answer, we recommend one in particular. Since what we mean by robustness is related to stability of "performance" (which is, for these examples, Mean Square Error MSE), it might prove useful to the user to provide a bound on the change in performance as the covariance moves about the parameter surface.

The total amount of change in MSE is the measure that will make a scheme robust or not. For both approaches, the stationary and non-stationary scheme, it is the maximum distance between two perturbations multiplied by the ninety five percent confidence bound. It is the ability of one to surely state, with ninety five percent confidence, that the scheme is robust or not. To compute such a bound one simply employs a ninety five percent confidence bound on the slope (calculated from the appropriate distribution) and then multiplies by the maximum distance one can

Table XII. Means, harmonic mean, medians, and 95 percentile confidence bound for each case scenario for the stationary case

Case	Mean	Harmonic Mean	Median	95%
1	0.783702	0.681948	0.848584	0.802822
2	0.789037	0.673989	0.858595	0.808419
3	0.783448	0.681552	0.847828	0.802609
4	0.817910	0.715011	0.884652	0.837573
5	0.816654	0.637963	0.893202	0.837647
6	0.774322	0.675513	0.841071	0.793107
7	0.856068	0.677125	0.948261	0.878800
8	0.783702	0.681948	0.848584	0.802822

Table XIII. Means, harmonic mean, medians, and 95 percentile confidence bound for each case scenario for the non-stationary case

Case	Mean	Harmonic Mean	Median	95%
1	0.830628	0.799697	0.881917	0.838295
2	0.830628	0.799697	0.881917	0.838295
3	0.830628	0.799697	0.881917	0.838295
4	0.827619	0.79686	0.881917	0.835318
5	0.839663	0.808334	0.881917	0.847204
6	0.831031	0.800077	0.881917	0.838693
7	0.839422	0.808101	0.881917	0.846967
8	0.840502	0.809145	0.881917	0.848029

travel between any two points on the parameter surface.

$$\Delta MSE\Big|_{max} = Maximum\ distance\ *\ (95\%\ Confidence\ Bound) \tag{5.24}$$

a. Maximum Distance for the Stationary Case

The maximum distance between two perturbations (ellipsoid of non circular radius), represented in figure 72, is difficult to compute. We first need to compute the maximum distance between two points in a ellipse of equation $C^2 + 2B^2 = \epsilon^2$ (see figure 73).

$$Max\ distance = \int_{-\frac{\epsilon}{\sqrt{2}}}^{\frac{\epsilon}{\sqrt{2}}} \sqrt{1 + (\frac{\partial C}{\partial B})^2} dB \tag{5.25}$$

This integral is called an *elliptic integral* and can be numerically computed [80]. In integral calculus, elliptic integrals originally arose in connection with the problem of giving the arc length of an ellipse. In the modern definition, an elliptic integral is any function f which can be expressed in the form:

$$f(x) = \int_c^x R(t, P(t))dt \tag{5.26}$$

where R is a rational function of its two arguments, P is the square root of a polynomial of degree three or four (a cubic or quartic) with no repeated roots, and c is a constant. In general, elliptic integrals cannot be expressed in terms of elementary functions. Exceptions to this are when P has repeated roots, or when $R(x, y)$ contains no odd powers of y. However, with appropriate reduction formula, every elliptic integral can be brought into a form that involves integrals over rational functions, and the three canonical forms (i.e. the elliptic integrals of the first, second and third kind).

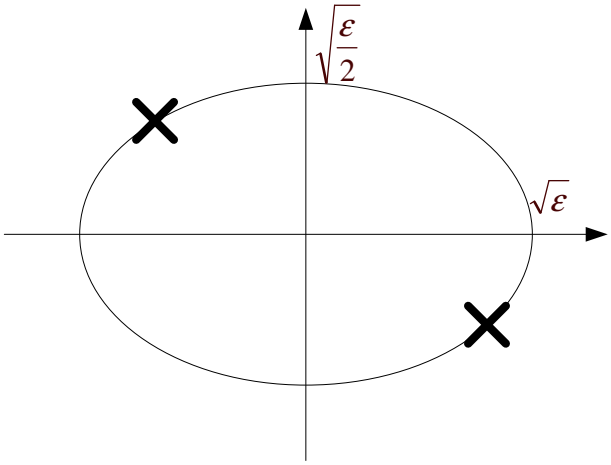

Fig. 72. Maximum distance between two points for the stationary case

The partial derivative of C with respect to B is equal to:

$$\frac{\partial C}{\partial B} = \frac{-2B}{\epsilon^2 - 2B^2} \tag{5.27}$$

Hence, the total amount of change in α is equal to:

$$\begin{aligned} \Delta MSE\Big|_{max} &= 2\int_0^{\frac{\epsilon}{\sqrt{2}}} \sqrt{1 + (\frac{\partial C}{\partial B})^2} dB * (95\%\ Confidence\ Bound) \\ &= 2\int_0^{\sqrt{\frac{\epsilon}{2}}} \sqrt{1 + (\frac{-2B}{\sqrt{\epsilon^2 - 2B^2}})^2} dB * (95\%\ Confidence\ Bound) \end{aligned} \tag{5.28}$$

The maximum distance between two perturbations (ellipsoid of non circular radius) for the stationary approach can be approximated to $\frac{\pi}{2} * \left(\epsilon + \frac{\epsilon}{\sqrt{2}}\right)$. Hence the total amount of change in MSE is then equal to:

$$\Delta MSE\Big|_{max} = \frac{\pi}{2} * \left(\epsilon + \frac{\epsilon}{\sqrt{2}}\right) * (95\%\ Confidence\ Bound) \tag{5.29}$$

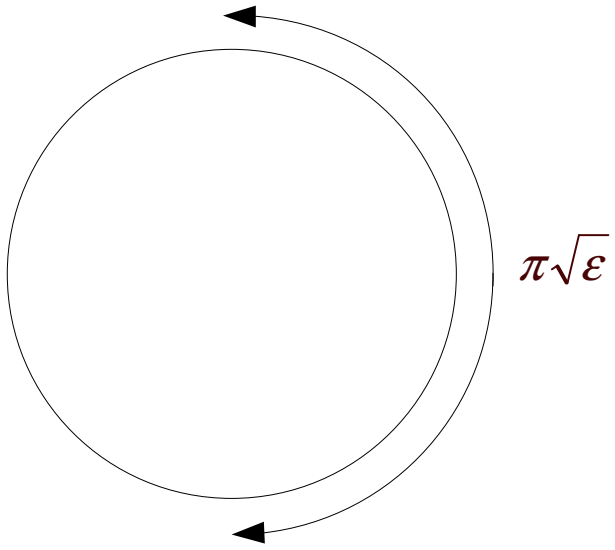

Fig. 73. Maximum distance between two points for the non-stationary case

b. Maximum Distance for the Non-Stationary Case

The maximum distance between two perturbations (sphere in 6D) for the non-stationary approach is equal to $\pi * \epsilon$ and is represented in figure 73. Hence the total amount of change in MSE is then equal to:

$$\Delta MSE\Big|_{max} = \pi * \epsilon * (95\%\ Confidence\ Bound) \tag{5.30}$$

3. ϵ_{max} at specified $\Delta MSE\Big|_{max}$

We remark that by making the parameter very close to the nominal by controlling ϵ to be small, $\Delta MSE|_{max}$ bound could be arbitrarily reduced. The comparison between stationary and non-stationary would translate into a comparison of the relative size of ϵ required to constrain the change in MSE (i.e $\Delta MSE|_{max}$) to be no more than a certain amount (i.e. 10% of the design goal of MSE). Since for both the non-stationary and stationary case the corresponding parameter surface is just a model

Table XIV. 95% confidence bound of $\Delta MSE\Big|_{max}$ for each case scenario for both the stationary and non-stationary approach

	$\Delta MSE\Big\|_{max}$	
Case	Stationary	Non-Stationary
1	0.374732	0.389172
2	0.377345	0.389172
3	0.374633	0.389172
4	0.397669	0.394452
5	0.39436	0.396701
6	0.737641	0.775817
7	0.417742	0.400431
8	0.749465	0.787382

of a ϵ-ball in a metric space of matrices, the dimensionality issue is alleviated. To compute such a bound one simply employs a ninety five percent confidence bound on the slope (calculated from the appropriate distribution) and then multiplies by the maximum distance one can travel between any two points on the parameter surface, which is a 2-D ellipsoid for the stationary case and a 5-D sphere for the non-stationary case. Since these distances (which even for the ellipsoid case, can be easily computed – see afore sections) involve ϵ, the result will lead to a maximal epsilon compatible with a specified $\Delta MSE|_{max}$ (in per cent) at ninety five percent confidence [56].

E. Conclusion

The chapter establishes a crucial nexus to our prior research: we develop methodology to answer the fundamental question, "Is the stationary case more robust than the non-stationary?" The chapter illustrates how, for example, gradient can be regarded as a random variable and a distribution generated by means of a density curve, allowing conclusions regarding robustness to be drawn from statistical metrics such as median, mean, harmonic mean and confidence bounds. In order for one to be able to surely state on the robustness of a scheme requires the computation of $\Delta MSE|_{max}$. This $\Delta MSE|_{max}$ is very dependent on the value of ϵ and is therefore directly related to the confidence one has in the covariance matrix. For example, the more confidence one has in the covariance matrix, the further away from the nominal values one can work, without destroying robustness.

This chapter also proves to go against expectations. One would have been expecting the simulations would prove that the stationary scheme to be much more robust (due to fact that there is more constraint on the stationary approach than the non-stationary). The results show that if one compares stationary to non-stationary

noise, robustness is definitely reduced by admitting non-stationarity. The effect, however is not dramatic, and so little may be lost by assuming the convenient stationarity. This research also uncovers some examples "swimming across the current". On the contrary we also see from case 7 (see tables XI, XII, XIII, and XIV) that robustness is marginally better for the non-stationary case, per some metrics.

Thus, less deviation from the nominal is allowed for the non-stationary case and it can therefore be judged to be less robust (although the differences are small). While the presence of non-stationary data compromises robustness, the amount of compromise is not large and may be considered acceptable - in view of the convenience in practice the assumption of stationarity offers. The aforementioned work has not only presented a versatile method for investigating robustness for a variety of applications, but has also addressed the specific question of the non-stationarity in the prediction context. As might have been expected, the admission of non-stationarity data has been seen to compromise robustness, but as perhaps not expected, the degree of compromise is quite small in all cases considered.

CHAPTER VI

ROBUST ALGORITHMS RELATING TO RESELECTING CELLS IN A CELLULAR WIRELESS COMMUNICATIONS SYSTEM

The present research and development relates generally to methods, devices and systems for reselecting and then handing over a mobile communications device from a first cell to a second cell in a cellular wireless communications system. More particularly, although not exclusively, aspects and embodiments of the algorithm relate to criteria for selecting a second cell while a mobile station is camped on, or otherwise interacting with and/or controlled by, a first cell. Particular aspects and embodiments of the present algorithm are well suited for use in a cellular wireless communications system which supports packet switched communications, for example according to the General Packet Radio Service (GPRS) standard, but are not limited to such an application.

A. Research and Background

It is well known that cellular wireless communication systems generally comprise a number (often large) of radio transceivers, or base stations, that define service areas or cells. The schematic diagram in figure 74 of the accompanying drawings, illustrates a system 100 comprising four base stations 120 defining respective cells 110. The cells typically overlap in order to ensure continuous coverage of service in the service areas. This is desirable for many reasons, not least because cellular systems are designed specifically to accommodate users as they move around within the system. In principle, mobile communications devices 130 interact with various base stations as the devices move through the respective cells 110 of the system 100.

One of the goals of a cellular wireless communication system is to enable a mobile

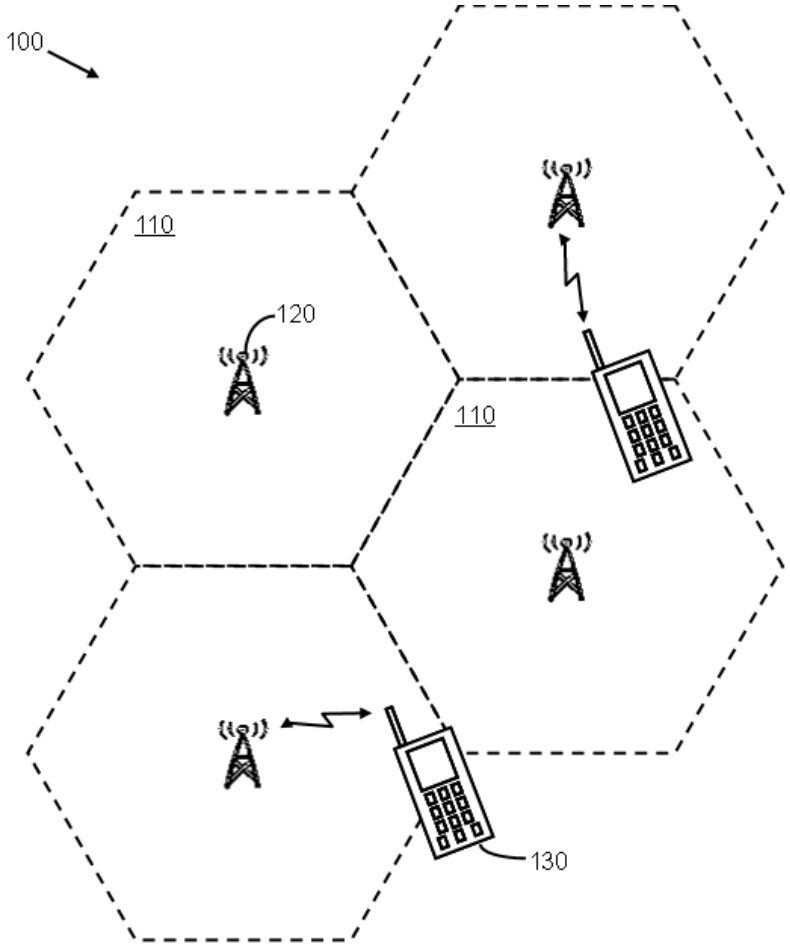

Fig. 74. Cellular wireless communication system

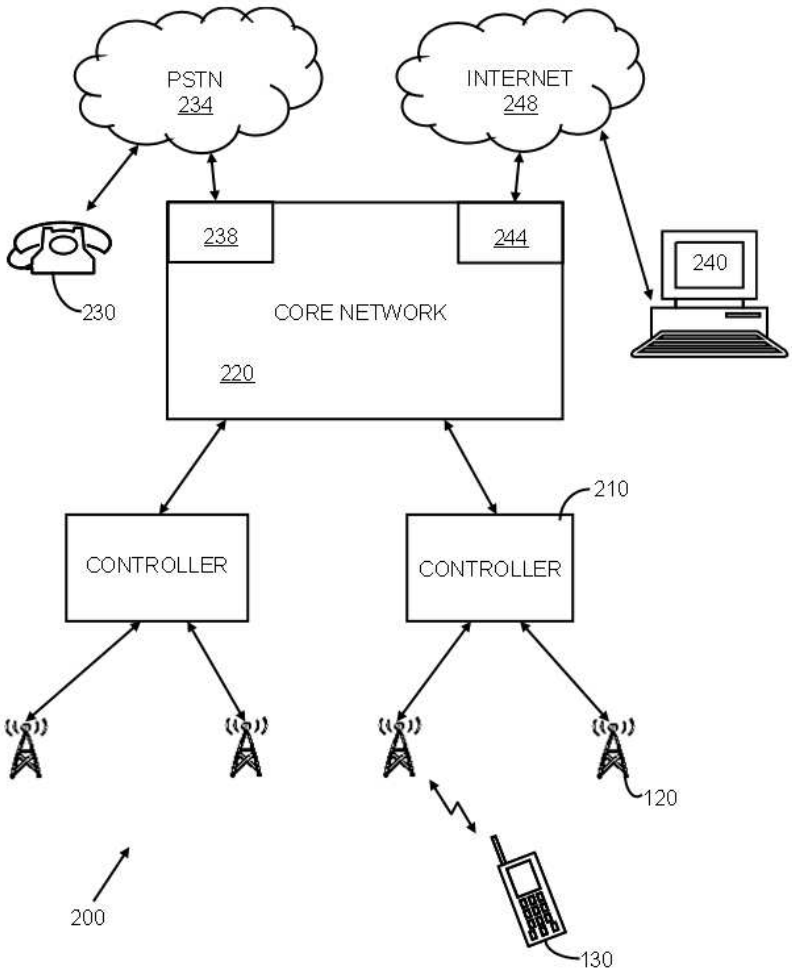

Fig. 75. Typical cellular network (main components in a cellular wireless communications system)

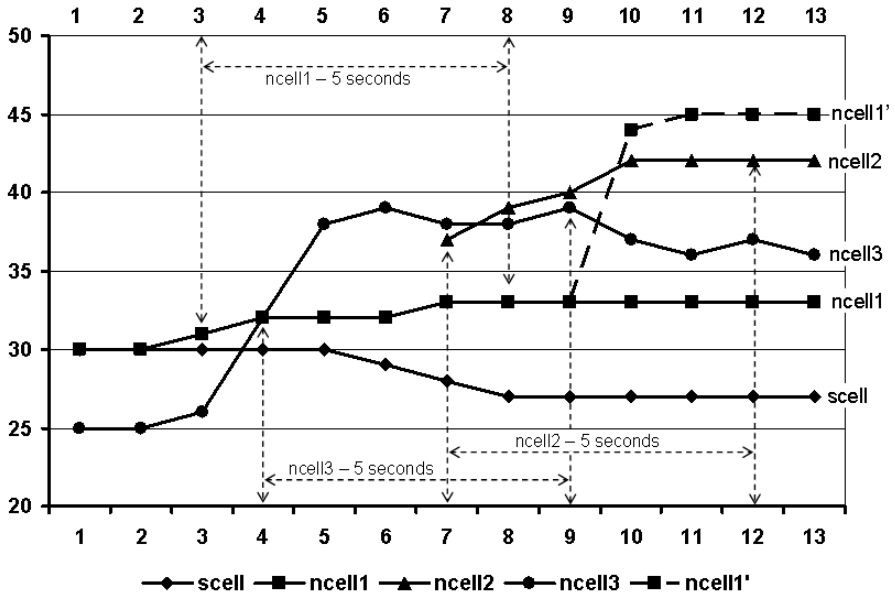

Fig. 76. Signaling parameter c2 levels for a serving cell and neighbor cells

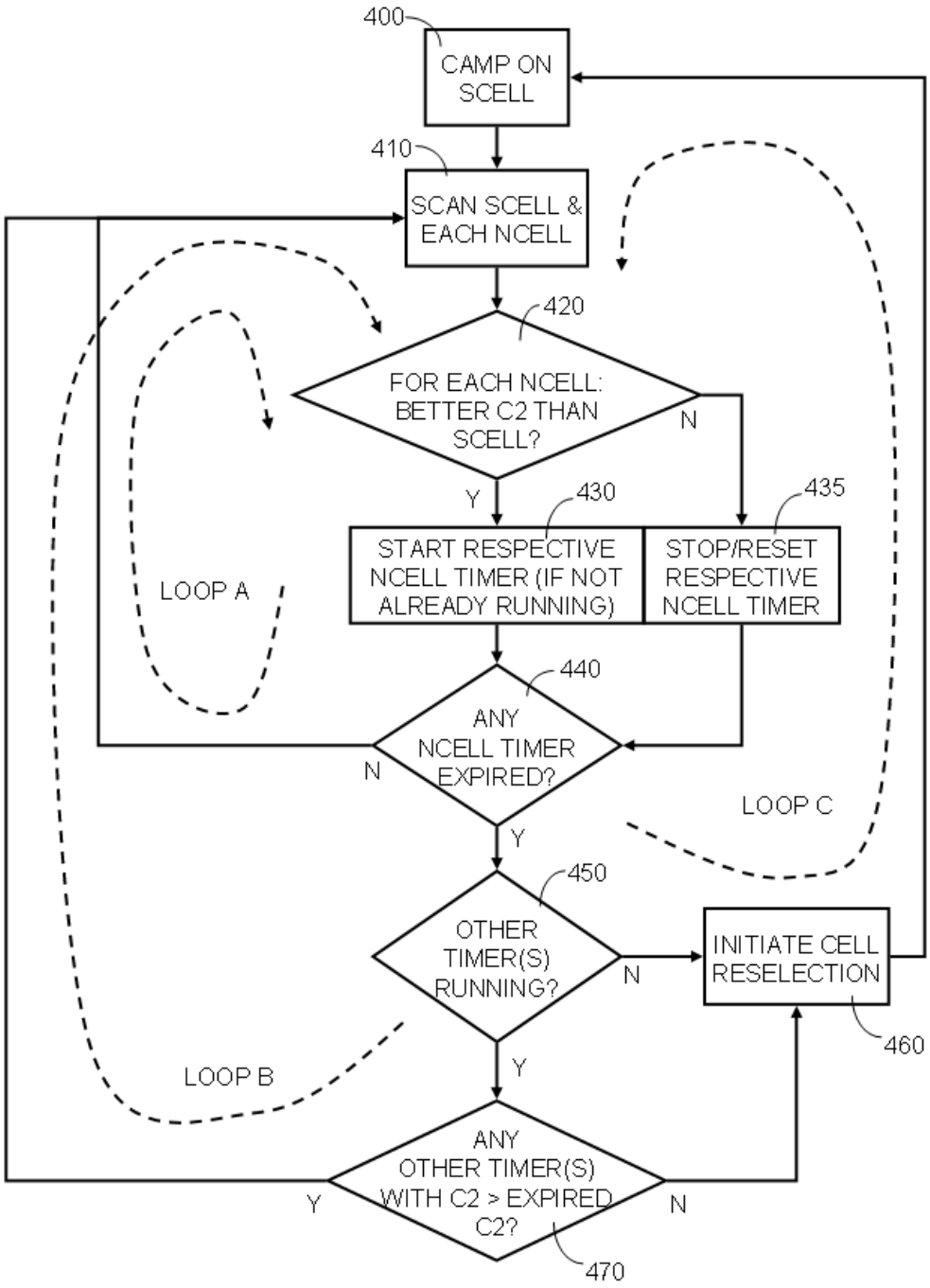

Fig. 77. Algorithm

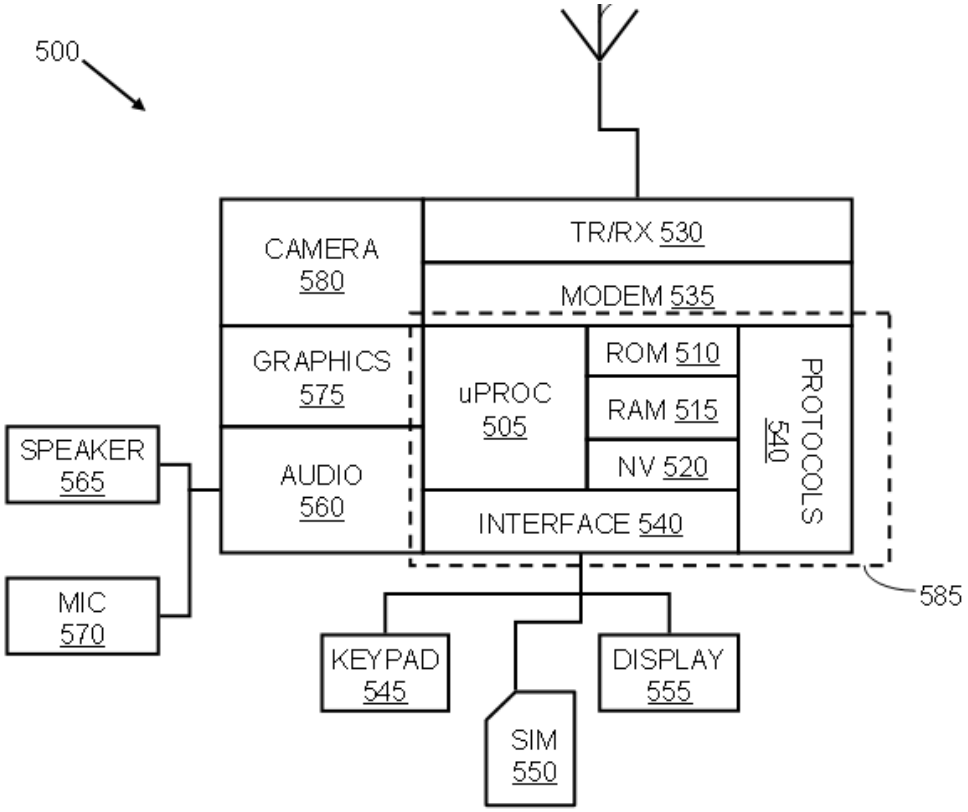

Fig. 78. Typical mobile station

communications device, which will be referred to herein for convenience as a mobile station, to remain connected to the system even when the user is moving through the system from one cell to another. Traditionally, the mobile station has been a so-called mobile phone or cellular phone, although, with advances in technology, a mobile station may be any one or more of a wide range of devices from solely voice devices to solely data devices. A mobile station may be anything from a traditional radio pager or mobile phone, though faxes, personal data assistants (PDAs), and music players, to computers, or any combination of these. This list is, of course, far from exhaustive. Indeed, although the term mobile station is used herein, the term is also intended to encompass devices that may not be user-operated or even user-operable, for example the device could be a wireless data card or the like, which is within another kind of apparatus.

Early cellular systems were circuit switched systems. That it to say, for each call the system created a circuit that reserves a channel for the user for the duration of the call. This is an inefficient use of resources, especially for bursty data. As technology has advanced, newer cellular systems have moved away from circuit switching to packet switching in which bursts of data are sent only when needed. Consequently, cellular systems have become more suitable for the transmission of data, which tends to be transmitted in bursts rather than a continuous stream.

As already mentioned each cell in a cellular system is defined and served by a base station. As a mobile station is moved from the service area defined by one cell into that defined by another, the system and the mobile station must break the connection with one base station and establish a connection with another base station whilst minimizing the connection loss between the mobile station and the system. This operation is sometimes known as a cell re-selection, a handoff or a handover. For simplicity of description only herein, the term re-selection will be used as a generic

term to describe the operations involved with a mobile station or equivalent moving from operating with one base station to operating with another base station; and the reader should import an alternative term, such as handover, handoff or the like, if the context so dictates. The term camped on is commonly used, and will be used hereinafter, to describe the base station with which, and respective cell in which, a mobile station is operating. That is, a cell re-selection involves a mobile station moving from being camped on one cell to being camped on another cell.

Typically, a cell re-selection can be initiated either by the mobile station or by the cellular system. How re-selection is initiated can depend on factors such as the kind of cellular system, its mode of operation and on the capabilities of a mobile station. In any event, re-selection is typically initiated either as a result of a service degradation, which tends to lead to increased power consumption requirements, or there being an opportunity to improve the service, which would lead to reduced power consumption requirements. Especially since many mobile stations operate from battery power, an opportunity to reduce power consumption, thereby improving power efficiency, is usually advantageous. Service degradation can result from factors such as increasing distance between a mobile station and a base station or natural or man-made obstructions such as hills or buildings respectively.

One known kind of re-selection operation requires a mobile station to monitor the signaling level and suitability of cells that neighbor the cell on which the mobile station is camped, which will be referred to hereinafter as the serving cell, and compare the monitored service levels with the signaling level and suitability of the serving cell. Then, if the signaling level and suitability of a neighboring cell is deemed by the mobile station to be better than that of the serving cell, for at least a predefined period of time (say, five seconds), the mobile station initiates a re-selection to the respective neighboring cell, which becomes the new serving cell. Such an operation

is described in §6.6.2 of ETSI Technical Specification document [81].

B. Prior Art and Problem Statement

After having conducted several field tests, controlled laboratory simulations and applied research, the present researchers [82, 83] have appreciated that, according to the prior art, unnecessary cell re-selection operations can occur. Since cell re-selection operations can consume a significant amount of power and/or result is a significant break in communications during an established voice call or other connection, we have appreciated that it would be advantageous to attempt to avoid unnecessary cell re-selection operations. Aspects and embodiments of the algorithm are, therefore, aimed at avoiding unnecessary cell re-selection operations.

C. Development

According to one aspect of the present algorithm there is provided a cell re-selection method for handing a mobile station from a serving cell to a selected target cell in a cellular wireless communications system, comprising plural cells, including the serving cell and plural other cells that are potential target cells, the method comprising: monitoring a first indicator, which is indicative of a signaling level of the serving cell; monitoring second indicators, each one being indicative of a signaling level of one of the plural potential target cells; initiating a timer associated with a potential target cell, when the respective second indicator indicates that the signaling level of the potential target cell is better than the signaling level of the serving cell, each timer having an associated expiry; if at least one timer has been initiated by the expiry of another initiated timer, then, after delaying for an additional period after at least the first timer has expired, selecting from the potential target cells a target cell having a

second indicator, which indicates that the signaling level of the respective target cell is better than the signaling level of the serving cell; and initiating a cell reselection of the mobile station to the selected target cell.

According to another aspect of the present algorithm there is provided a cellular wireless communications system comprising plural cells, including a serving cell and plural potential target cells, and a mobile station operable with the cellular wireless communications system, the system comprising: a first process, arranged to monitor a first indicator, indicative of a signaling level of the serving cell; a second process, arranged to monitor second indicators, each one being indicative of a signaling level of one of the plural potential target cells; a third process, arranged to compare the first indicator with the second indicators and initiate a timer associated with a potential target cell whenever the respective second indicator of the potential target cell is better than the first indicator, each timer having an associated expiry; and a fourth process, arranged to select from the potential target cells, if at least one timer has been initiated by the expiry of another initiated timer and after delaying for an additional period after at least the first timer has expired, a target cell having a second signaling indicator, which indicates that the signaling level of the respective target cell is better than the signaling level of the serving cell, and initiate cell reselection to the selected target cell.

According to yet another aspect of the present algorithm there is provided a mobile station adapted for operation in a cellular wireless communications system, the device comprising: a receiver arranged to receive signals and derive indicators therefrom, each indicator being indicative of the signaling level of a cell of the system; and a processor arranged to operate a cell re-selection operation, the operation comprising: a first process, arranged to monitor a first indicator, indicative of a signaling level of the serving cell; a second process, arranged to monitor second indicators, each one

being indicative of a signaling level of one of the plural potential target cells; a third process, arranged to compare the first indicator with the second indicators and initiate a timer associated with a potential target cell whenever the respective second indicator of the potential target cell is better than the first indicator, each timer having an associated expiry; and a fourth process, arranged to select from the potential target cells, if at least one timer has been initiated by the expiry of another initiated timer and after delaying for an additional period after at least the first timer has expired, a target cell having a second signaling indicator, which indicates that the signaling level of the respective target cell is better than the signaling level of the serving cell, and initiate cell re-selection to the selected target cell. The algorithm also provides a communication device in which characteristics of signals of different sources are monitored over respective time periods and a source is selected for communication when the monitored characteristic has satisfied certain criteria over substantially all of its respective time period. The algorithm extends to a radio communication unit in which signals of different cells are monitored in time periods and a cell is selected for communication when the monitored signal thereof has satisfied certain criteria by the end of its respective time period.

The algorithm also extends to a transceiver in which signals of different communications nodes in a communications network are observed during associated intervals to determine a signal behavior which is used identify at least one node suitable for subsequent communication with the transceiver. The above and further features of the algorithm are set forth and together with advantages thereof will become clearer from consideration of the following detailed description of an embodiment of the algorithm given by way of example with reference to the accompanying drawings.

D. Brief Description of Figures

In the drawings: Figure 74 is a schematic diagram showing a cellular wireless communications system; Figure 75 is a high level block diagram showing the main components in a cellular wireless communications system; Figure 76 is a graph showing a comparison of signaling parameter C2 levels for a serving cell and neighbor cells; Figure 77 is a flow chart illustrating a cell re-selection operation according to one embodiment of the present algorithm; and Figure 78 is a block diagram showing the main functional components of a typical mobile station that may be configured to operate in accord with embodiments of the present algorithm.

E. Detailed Description

Turning now to the schematic diagram in Figure 75 of the accompanying drawings, there is shown a high level block diagram of a typical wireless cellular communications system, for example as shown in figure 74. For the purposes of illustration, the system includes only four base stations 120, which provide access to the system for mobile stations 130, for example mobile telephone handsets. Each base station 120 is controlled by a controller 210 and each controller 210 is connected to a core network 220 of the system, via an appropriate communications infrastructure. Each controller 210 can control one base station 120 but typically a controller controls more than one base station. The core network 220 in general contains the infrastructure, components and functionality for controlling the controllers 210; routing calls and connections of all kinds from and to mobile stations 130; routing calls and connections from mobile stations 130 to other systems and terminating equipment; and receiving calls and connections, from other systems and terminating equipment, which are intended for mobile stations. Examples of other terminating equipment are traditional telephone

equipment 230, which are connected to via a PSTN 234 and PSTN gateway 238 of the core network 220, Internet servers 240, which are connected to via an Internet gateway 244 and the Internet 248, and other telecommunications systems or services (not shown), such as voice-mail or corporate networks respectively.

There are various kinds of wireless cellular communications systems, which operate according to various different standards. Such systems and standards include, but are not limited to, GSM, GPRS and third generation standards such as UMTS and WCDMA. The diagram in figure 75 is intended to be generic, and apply, at least functionally, to all such standards and systems. Particular embodiments of the present algorithm relate to cell re-selection in a GPRS system. According to the GPRS terminology, a base station 120 is commonly referred to as a base transceiver station (BTS) and the controller 210 is commonly referred to as a base station controller (BSC). The combination of BTS and BSC is commonly referred to as the base station subsystem (BSS). Hereafter, while GPRS components will be referred in order to describe particular embodiments of the present algorithm, it will be appreciated that the principles taught apply equally to other kinds of wireless cellular communications systems, such as GSM and 3G.

Turning now to figure 76, the graph therein shows four trend lines, which illustrate exemplary signaling relationships between a mobile station, its serving cell (scell), on which the mobile station is camped, and three neighboring cells (ncell1-ncell3). The signaling relationships are characterized by a signaling parameter, C2, the value of which provides an indication of the strength of signals received by the mobile station from the respective cells. In general, a higher value of C2 indicates a stronger signal between the mobile station and the cell.

It will be evident that alternative embodiments of the present algorithm may rely on deriving and/or monitoring different indicators, parameters and characteristics of

systems and mobile stations, insofar as the parameters and characteristics relate in some way to the likely signaling performance or capability between the mobile station and a base station or similar.

In known GPRS systems, a mobile station monitors the C2 values of all cells that are classed as neighbors of the serving cell. Each cell identifies which other cells are classed neighbors and a list of neighbors is communicated to a mobile station, by a new serving cell, during or soon after a cell re-selection operation. The mobile station attempts to monitor the C2 values for the serving cell and all neighbor cells during the time the device is camped on the serving cell. A mobile station generates C2 values in a pre-defined way, for example as described in §6.4 of the aforementioned ETSI document, by evaluating various characteristics of signals received from the neighbor cells; although the most important characteristic is typically signal power. Typically, a mobile station will scan for neighbor cell signals periodically, for example every second, or as otherwise defined by a control program of the mobile station, in order to monitor the C2 levels.

Referring to the trend lines in the graph in figure 76, it is shown that the C2 value of scell decreases over time, from a value of 30 to a value of about 27. After two seconds, the C2 value for ncell1 becomes higher than the C2 value of scell, and remains higher for the remainder of the period shown. This may be as a result of the mobile station moving away from the scell BTS and towards the ncell1 BTS. In principle, it would appear to make sense for the mobile station to select ncell1 as a new serving cell in order to improve power-efficiency. In practice, this is exactly what happens according to prior art re-selection operations. Specifically, according to the prior art, as soon as the mobile station detects that the C2 value of ncell1 is higher than the C2 value of scell (which, according to the graph, is when three seconds have lapsed), the mobile station starts a timer running. Then, if, after the timer expires

(for example after another five seconds), the situation remains the same, at around eight seconds, a re-selection to ncell1 is initiated by the mobile station.

According to the graph in figure 76, by four seconds, the C2 value for ncell3 rapidly becomes higher than the C2 value of scell. Indeed, after four seconds, the C2 value of ncell3 also exceeds the C2 value of ncell1, and remains higher for the remaining period shown. Such a dramatic increase in the C2 power of ncell3 may result from, for example, the mobile station moving out from behind an obstruction that was inhibiting receipt of transmissions from the BTS of ncell3. Furthermore, at around seven seconds, the C2 value of ncell2 suddenly increases. The suddenness of the increase may be because, up until seven seconds, attempts to scan signals from that neighbor cell had failed. Similarly, the sudden increase may be due to the mobile station moving out from behind an obstruction that was blocking receipt of transmissions from the BTS of ncell2.

According to the graph in figure 76, applying a known re-selection operation, re-selection from scell to ncell1 is initiated after eight seconds, even though, at that point in time, the C2 level of both ncell2 and ncell3 exceeds the C2 level of ncell1. The result is that, as soon as cell reselection is complete, and the mobile station begins to monitor C2 levels of its new neighboring cells (assuming at least one of ncell2 and ncell3 is a neighboring cell of ncell1), another timer is triggered almost immediately with a view five seconds later to initiating another reselection operation to one of ncell2 or ncell3.

A cell reselection operation, by its nature, can cause a significant disruption to communications. This is at least in part because, according to the GPRS standard, as soon as a mobile station reselects to a new cell, the mobile station can spend as long as eight seconds reading broadcast information before camping onto the new cell.

The present researchers have appreciated that it can be inefficient to initiate

cell reselection to the first neighbor cell that appears to have an improved C2 value. The graph in figure 76 illustrates this very well, wherein, according to the known reselection operation, the mobile station reselects to a cell, ncell1, which does not have the best C2 value at the time of reselection.

An embodiment of the present algorithm will now be described in detail, wherein, in a GPRS system, a cell reselection operation is held off, in other words delayed, until it is evident that reselection will be to the most appropriate neighbor cell. It will, however, be appreciated that the principles are equally applicable in GPRS systems and in other kinds of wireless cellular communications systems, wherein the reselection may be initiated by the BSS (or equivalent), the core network or by a mobile station.

Referring now to the flow diagram in figure 77, in a first step 400, a mobile station becomes camped on a new serving cell (scell) and acquires its necessary upstream signaling information and neighbor cell information from the respective BTS, in a known way. In a next step 410, the mobile station scans for signals from the serving cell and the respective neighboring cells and derives the respective C2 values. In a next step 420, the mobile station determines whether a C2 value of any neighboring cell is better than the C2 value of the serving cell. If the result of the determination is positive for any neighboring cell (that is, the neighboring cell has a higher C2 value than the serving cell), then, in a next step 430, the mobile station starts a timer, which the mobile station associates with the respective neighbor cell. If a respective timer is already running then no additional action occurs and the timer is left running. If, on the other hand, the result of the determination is negative for any particular neighboring cell (that is, the neighboring cell has a lower C2 value than the serving cell), then, in a step 435, any respective running timer is stopped and reset. If no timer is running, then no action occurs.

In other words, according to the present exemplary embodiment, for each scan operation, a timer is started (or permitted to continue) for any neighboring cell which has a better C2 value than the source cell. A timer runs until a pre-determined expiry time, unless, before or on expiry, the C2 value drops below the C2 value of the source cell, in which case the timer is stopped and reset (or otherwise cancelled). Obviously, in the example provided, a timer is not started for a neighboring cell unless its C2 becomes better than the C2 value of the source cell.

Next, in a step 440, the mobile station determines whether any timer has expired. In this example, the timers are set to expire after five seconds. In other examples, a different expiry time may be selected. In the step 440, if no timer has expired, then the process returns to the step 410, whereat the mobile station again scans for signals from the source cell and neighbor cells, and the process repeats.

If, however, in the step 440, the mobile station establishes that a timer has expired, the mobile station enacts a further check, in a step 450, to determine whether any other timers are running. If no other timers are running then, according to this embodiment, the mobile station initiates a cell reselection operation, in a step 460, to the neighboring cell, to which the expired timer belongs. If two or more timers expire at the same time then the cell reselection operation reselects to the neighboring cell that has the best C2 value. The process then repeats from the step 400.

If, in the step 450, the mobile station determines that at least one other timer is running, the process enacts a final check, in a step 470, to establish if the C2 level of the (or at least one) cell, for which a timer is running, is better than the C2 level of the cell (or cells, if more than one has expired) associated with an expired timer. If the result of the check in step 470 is negative (that is, of the timers still running none is associated with a neighboring cell that has a better C2 value than the or each timer that has expired), then the process jumps to step 460, where a cell reselection

operation takes place, to reselect to the neighbor cell that has expired and has the highest C2 value.

If the result of the check in step 470 is positive (that is, of the timer(s) running, at least one is associated with a neighboring cell that has a better C2 value than each neighboring cell for which a timer has expired) the process jumps back to the step 410, where at the mobile station holds-off reselection, scans for signals from the source cell and the neighboring cells, and the process repeats.

In other words, according to the present exemplary embodiment, for each scan operation, cell reselection is held-off (or otherwise delayed) as long as at least one timer is running and the C2 value of the neighboring cell, which is associated with that timer, is better than the C2 value of neighboring cells for which an associated timer has already expired.

The cell reselection operation will now be tested against the C2 level information shown in the graph in figure 76. As can be seen in the flow diagram in Figure 77, the reselection operation has three main decision loops, identified as A, B and C. It is evident that, up until eight seconds on the graph in figure 76, before any timer expires, the operation executes only the process steps in loop A, due to the test in step 440. While iterating around loop A, as a result of the test in the step 420 returning a positive result, three timers are started; at three seconds (for ncell1), four seconds (for ncell3) and seven seconds (for ncell2). At eight seconds, however, the first timer (for ncell1) expires and the test in step 440 becomes positive. Since, two other timers (for ncell3 and ncell2) are still running at eight seconds, and both have a better C2 value than the cell (ncell1) for which the timer has expired, the process enters loop B, due to the positive result of the test in the step 450 (which indicates that other timers are still running) and the positive result in step 470, which indicates that at least one cell for which a timer is still running has a better C2 value than the cell (or

cells) for which the timer has (or timers have) expired.

In contrast, in the prior art reselection operation, loop B does not exist, since reselection occurs as soon as a first timer expires. From the point where a first timer expires, the process remains in loop B.

According to the graph in figure 76, the first occasion when a timer expires (for ncell2), and no other timer is running (which is associated with a neighboring cell which has a better C2 value), is at 12 seconds, where at the process enters loop C, as a result of the negative result of the test in the step 450, and reselection to the cell having the last timer to expire and the best C2 level (that is, ncell2) is initiated in step 460.

In the foregoing exemplary embodiment, it is clear that the last timer to start is associated with ncell2. It is also clear that, at twelve seconds, ncell2 also has the best C2 level. Consequently, the operation reselects to ncell2. This process avoids at least one unnecessary cell reselection operation.

In other instances, according to exemplary embodiments of the present algorithm, the last timer to start before reselection occurs might not be associated with the neighbor cell that has the best C2 value when the last timer expires. For example, referring again to the graph in figure 76, assume that the C2 level of an alternative ncell1 increases rapidly for some reason at around nine seconds, and exceeds the C2 value of ncell2 at around ten seconds, as illustrated by the broken ncell1 trend line in the graph. In this alternative example, since the ncell1 value was already higher than the scell C2 value at nine seconds, no new timer would have been started. Indeed, the ncell1 timer had previously started and then expired by eight seconds. According to this example, as a result of a negative test result in step 470 of the process, a reselection to alternative ncell1 would take place at around ten seconds, even though the timer associated with ncell2 would not otherwise expire for another two seconds.

A similar, advanced reselection would occur in the event that the C2 value of any cell for which a timer had expired increased above the C2 level of all cells having running timers.

In the first of the foregoing examples, cell reselection is held off for a further four seconds from eight seconds to 12 seconds - after the first timer expires. As a result, a needless reselection from ncell1 to ncell2 is avoided. In the second example, in fact, a reselection to the alternative ncell1, ncell1, is merely delayed from eight seconds to ten seconds, which is not a significant delay in practical terms. It is anticipated that, on the whole, circumstances similar to those of the first example would occur far more often than those of the second example.

The functional components of an exemplary mobile station 500 are illustrated in the block diagram in figure 78. The device in this example might be a mobile telephone handset. Embodiments of the present algorithm can be enacted by such a device. The device generally comprises an embedded processor 505, for controlling the overall operation of the device 500. The processor 505 has associated memory, including ROM 520, RAM 515 and non-volatile memory 520, for example for storing a control program of the device, application programs and/or an address book. Some or all of the memory might be separate from the processor. The device includes an antenna 525, which is connected to transmit/receive circuit 530, which communicates signals to and from the processor 505 via a modem 535. The device is arranged to interact with a base station according to several protocols, for example GSM, GPRS and/or 3G, which are supported by respective application programs, which are typically stored in a protocol module area 540 of non-volatile memory of the device. An interface module 540 facilitates communications with a keypad 545, a subscriber identity module (SIM) 550 and a display screen 555 of the device. An audio module 560 supports a speaker 565 and a microphone 570. A graphics processor

575 is included for processing graphics, for example for display on the display screen 555 and, in this example, the device includes a camera module 590.

A device of the kind shown in figure 78 is generally known in the prior art and it is typically an application program that needs to be arranged to control the device to operate in accord with embodiments of the present algorithm. For example, in embodiments of the present algorithm that operate in a GPRS cellular wireless communications system, a GPRS application program is arranged to operate generally in accord with the flow diagram of figure 77, at least insofar as a cell reselection operation is concerned. In any event, at least a subset of the main components of the device in figure 78, as shown within the dotted line 585, may be provided as a single chip device, or as plural chips or components, which can be installed in a mobile station to operate according to embodiments of the present algorithm.

It will be appreciated that cell reselection can be held off and then initiated in many different ways, all within the scope of aspects and embodiments of the present algorithm.

F. Conclusion

Cell reselection comprises monitoring a first indicator indicative of a signaling level of the serving cell, and monitoring second indicators, each indicative of a signaling level of one of plural potential target cells. A timer associated with a potential target cell is initiated when the respective second indicator indicates that the signaling level of the potential target cell is better than the signaling level of the serving cell. Each timer has an associated expiry, and, if at least one timer has been initiated by the expiry of another initiated timer, then, after delaying for an additional period after at least the first timer has expired, a target cell is selected from the potential target

cells. The target cell is selected that has a second indicator which indicates that the signaling level of the respective target cell is better than the signaling level of the serving cell.

CHAPTER VII

CONCLUSIONS

A. Dissertation Summary

Engineering has always had to deal with inexact knowledge in both design and analysis. Certain assumptions are made, resulting in so-called optimal algorithms, but the fidelity of such algorithms is limited by the accuracy of the assumptions. Accordingly, often more useful algorithms for the practitioner are those whose performance is less sensitive to the inexact knowledge. This reduced sensitivity is associated with robust procedures, and the study of robustness is an important area of research into algorithms, which have the potential to not fail in the field.

The thrust of the work proposed in our dissertation is to extend the differential geometric approach toward robustness in several important ways. In this research, we consider a system of a linear estimator (multiple tap filter) and then attempt to model the system performance and robustness in a graphical manner, which admits an analysis using the differential geometric concepts. We try to compare two different perturbation models, namely the slope with biased perturbations of a surface and the slope with unbiased perturbations, and observe the values to see which of them can alternately be used in the process of understanding or measuring robustness. We show that gradient can be viewed as a random variable and therefore used to generate probability densities allowing one to conclude on robustness. We also used the differential geometric methodology to the prediction of time varying deterministic data in imperfectly known non-stationary distribution.

We note that past work made use of both Euclidean space and curved space to model the inexact knowledge. Nevertheless, this past work has admitted non

stationary data only for Euclidean space. Since it is very likely that the real world will feature various degrees of non- stationarity, and since we have seen that often a curved space model is more appropriate, we applied the differential geometric approach to measure robustness in the scenario where the data is non stationary with a curved space model. We establish a crucial nexus to our research of stationary model and we address the fundamental question, "Is the stationary case more robust than the non-stationary?". We compared results with the robustness values obtained under a non-stationary assumption to see if residual stationarity degrades robustness. Robustness is marginally reduced by admitting non-stationarity.

Finally we have gone beyond the Halverson differential geometric approach to research information geometry and investigate applications toward robustness.

We then conducted applied research and developed a robust handoff algorithm, relating generally to methods, devices and systems for reselecting and then handing over a mobile communications device from a first cell to a second cell in a cellular wireless communications system. This algorithm results in significant decrease in amount of power and/or result is a decrease of break in communications during an established voice call or other connection, in the field.

B. Future Work

The above research can be extended in several ways. Some of the steps are listed below:

- Information geometry provides mathematical science with a new framework for analysis. Thus, transcending differential geometry and exhaustively pursuing application in Information Geometry would show, the techniques naturally connect with robustness. There is need to seek such a nexus and to either develop

it or to determine that it is unlikely that one exists. While not essential, success in this future effort could provide a useful confirmation of the efficacy of our approach.

- Since the unbiased model is preferable in terms of telling the "true" robustness story, our results show that the extra difficulty in computing unbiased gradient densities is worth the effort in the simulations domain. Nevertheless, we remark that unbiased approach involves a matrix inversion, G^{-1}, linear algebra operation. The complexity of matrix inversion in hardware becomes prohibitive for real time applications and large values of matrix dimension. This operation is inherently complex to design in hardware domain, more so in real time systems and requires fast, pipelined and scalable hardware architecture. Often iterative algorithms are used in such situations. Bridging the gap between these two highly specialized domains allows more degrees of freedom when determining the final solution.

- Our research employs gradient as first order measure. As indicated in the introduction, such an approach is not always entirely sufficient, and successful employment of curvature is undertaken for a flat space model. In addition curvature as well as gradient can be displayed as a distribution in three dimensions regardless of the dimension of the curved space model, thus raising the potential work for both geometric invariants to be useful tools for robustness measurement. Thus, there is the need for higher order tool with a curved space model.

REFERENCES

[1] S. Kassam and H. Poor, "Robust techniques for signal processing: A survey," *Proc. IEEE*, vol. 73, pp. 433–480, Mar. 1985.

[2] P. Huber, "Robust statistical procedures," in *CBMS NSF Regional Conference Series in Applied Mathematics*, 2nd Edition, Philadelphia, PA, Jan. 1996, pp. 5–12.

[3] P. Huber, *Robust Statistics*. vol. 1. Hoboken NJ: Wiley, Inc., 1981.

[4] M. Thompson, D. Halverson, and G. Wise, "Robust detection in nominally laplace noise," *IEEE Trans. Communications*, vol. 42, no. 3, pp. 1651–1660, Feb./Mar./Apr. 1994.

[5] A. El-Sawy and V. V. Linde, "Robust detection of known signals," *IEEE Trans. Info. Theory*, vol. IT-23, pp. 722–727, Nov. 1977.

[6] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer, 1994.

[7] P. Huber and V. Strassen, "Minimax tests and the Neyman-Pearson lemma for Capacities," *Ann. Statist*, vol. 1, no. 2 , pp. 251–263, Mar. 1973.

[8] D. Halverson, "Robust estimation and signal detection with dependent non-stationary data," *Circuits Systems and Signal Processing*, vol. 14, pp. 465–472, Jul. 1995.

[9] M. Thompson, D. Halverson, and C. Tsai, "Robust estimation of signal parameters with non stationary and/or dependent data," *IEEE Trans. Info. Theory*, vol. 39, pp. 617–623, Mar. 1993.

[10] M. Thompson and D. Halverson, "A differential geometric approach toward robust signal detection," *J. Franklin Inst.*, vol. 328, no. 4, pp. 379–401, Jan. 1991.

[11] M. Thompson and D. Halverson, "Geometric measures of robustness in signal processing," in *Advances in Communications and Signal Processing* (edited by W.A. Porter and S.C. Kakin), New York: Springer, 1989, pp. 348–358.

[12] W. Liu, D. Halverson, S. Akkihal, and M. Thompson, "Local and nonlocal robustness measures with applications to distributed sensor systems," *IEEE Trans. Aerospace Electronic System*, vol. 38, no. 2, pp. 675–681, Feb. 2002.

[13] C. Tsai and D. Halverson, "Average robustness in signal detection and estimation," *J. Franklin Inst.*, vol. 333 B, pp. 127–139, Jan. 1996.

[14] W. Liu and D. Halverson, "Robustness of the sign detector in dependent noise," *J. Franklin Inst.*, vol. 336, no. 7, pp. 1155–1174, Sept. 1999.

[15] M. Bakich and D. Halverson, "Non-Euclidian robustness measures for communications and signal processing," *in Proc. of the Eleventh Annual International Conference on Signal Processing Applications and Technology*, Oct. 16-19, 2000, pp. 567–601.

[16] F. Kellison and D. Halverson, "Applications of curvature toward the measurement of robustness for data processors," *Computational Statistics and Data Analysis*, vol. 11, no. 37, pp. 343–362, Sept. 2001.

[17] M. Bakich, *Non-Euclidean Robustness Measures for Communications and Signal Processing*. Ph.D. dissertation, Texas A&M University, 2002.

[18] V. V. Varma and D. Halverson, "Applications of unbiased perturbations towards

providing robustness with pragmatic geometric methods," *Computational Statistics and Data Analysis*, vol. 49, issue 4 , pp. 1228-1243, 15 Jun. 2005.

[19] K. Gauss, *Theory of Motion of Heavenly Bodies.* New York: Prentice Hall, 1963.

[20] S. Amari, *Differential Geometric Methods in Statistics, (Lecture Notes in Statistics).* Berlin, Germany: Springer, vol. 25, 1985.

[21] S. Amari, and H. Nagaoka, *Methods of Information Geometry.* American Mathematical Society: Oxford University Press, 2000.

[22] S. Amari and T. Han, "Statistical inference under multi-terminal rate restrictions:A differential geometric approach," *IEEE Trans. Info. Theory*, vol. 35, pp. 217–227, Mar. 1989.

[23] S. Amari, "Information geometry of statistical inference-An overview," *IEEE Information Theory Workshop*, Bangalore, India, Oct. 2002, pp. 86–89.

[24] S. Amari, "Information geometry on hierarchy of probability distributions," *IEEE Trans. Info. Theory*, vol. 47, no. 5, pp. 1701–1711, Jul. 2001.

[25] S. Amari, O. Tomoko and P. Hyeyoung, "Information geometry of adaptive systems," *The IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Alberta, Canada, 2000, pp 12–17.

[26] L. Campbell, "Minimum cross entropy estimation with inaccurate side information," *IEEE Trans. Info. Theory*, vol. 45, pp. 2650–2652, Nov. 1999.

[27] S. Ikeda, T. Tanaka, and S. Amari, "Information geometry of turbo codes," In *Proceedings of 2002 IEEE International Symposium on Information Theory (ISIT 2002)*, p.114, Lausanne, Switzerland, Jun. 2002.

[28] S. Ikeda, T. Tanaka, and S. Amari. "Information geometry of Turbo and LDPC codes," *IEEE Trans. Info. Theory*, vol. 50, no.6, pp. 1097–1114, Jun. 2004.

[29] L. L. Campbell, "The relation between information theory and the differential geometric approach to statistics," *Inform. Sci.*, vol. 35, pp. 199-210, Jun. 1985.

[30] S. Amari, "Differential geometry of a parametric family of invertible linear systems, Riemannian metric, dual affine connections and divergence," *Math. Syst. Theory*, vol. 20, pp. 53-82, Dec. 1987.

[31] S. Amari, "Fisher information under restriction of Shannon information," *Ann. Inst. Statist. Math.*, vol. 41, pp. 623-648, Dec. 1989.

[32] S. Amari, "Dualistic geometry of the manifold of higher-order neurons," *Neural Networks*, vol. 4, issue 4, pp. 443-451, 1991.

[33] S. Amari, "Information geometry of the EM and EM algorithms for neural networks," *Neural Networks*, vol. 8, no. 9, pp. 1379-1408, Jan. 1995.

[34] M. Szewczul, D. Halverson, and M. Thompson, "Modelling and measuring detector performance/robustness in nominally laplace noise," *International J. of Modelling and Simulation*, vol. 24, no. 1, pp. 20–25, Jan. 2004.

[35] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice Hall Signal Processing Series, 1998.

[36] S. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. Englewood Cliffs, NJ: Prentice Hall Signal Processing Series, 1998.

[37] E. Lehman, *Testing Statistical Hypotheses*. New York: Springer, 1959.

[38] R. McDonough and A. Whalen, *Detection of Signals in Noise*. New York: Academic Press, 1995.

[39] H. V. Trees, *Detection, Estimation, and Modulation Theory*, vol. I-III. New York: Wiley-Interscience, 1968-1971.

[40] S. M. Kendall and A. Stuart, *The Advanced Theory of Statistics*, vol. 2. New York: Hafner Publishing Company, 1979.

[41] G. Box and G. Tiao, *Bayesian Interference in Statistical Analysis*. Reading, MA: Addison-Wesley, 1973.

[42] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[43] E. Lehmann, *Testing Statistical Hypothesis*. New York, Wiley-Interscience, 1959.

[44] H. V. Trees, *Detection, Estimation, and Modulation Theory, Part I*. New York: Wiley, 1968.

[45] E. Lehman, *Testing Statistical Hypotheses*. New York: Springer, 1970.

[46] S. Chern and R. Osserman, "Lebnitz Integrals," *Proceedings of Symposia in Pure Mathematics: Differential Geometry*, vol. XXVII. Austin, TX: Advanced Mathematic Symposium, 1975.

[47] S. Waner, Lecture Notes "Introduction to differential geometry and general relativity." Hofstra University, Online available. http://people.hofstra.edu/faculty/StefanWaner/diffgeom/tc.html, April 19th 2005.

[48] H. V. Poor, "Signal detection in the presence of weakly dependent noise - part ii: Robust detection," *IEEE Trans. Inform. Theory*, vol. 28, pp. 744–752, Sept. 1982.

[49] G. V. Moustakides and J. B. Thomas, "Robust detection of signals in dependent noise," *IEEE Trans. Inform. Theory*, vol. 33, pp. 11–15, Jan. 1987.

[50] E. C. Martin and H. V. Poor, "On the asymtotic efficiencies of robust detectors," *IEEE Trans. Inform. Theory*, vol. 38, pp. 50–60, Jan. 1992.

[51] O. Kenny and L. White, "Robust detection of signal classes," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Detroit, USA, vol. 34, no. 1, pp. 3131–3133, May 1995.

[52] D. J. Warren and J. B. Thomas, "Asymptotically robust detection and estimation for very heavy-tailed noise," *IEEE Trans. Inform. Theory*, vol. 37, pp. 475–481, May 1991.

[53] J. Makhoul, "Linear prediction: A tutorial review," in *Proceedings of the IEEE*, vol. 63, pp. 561-580, Apr. 1975.

[54] G. U. Yule, "On a method of investigating periodicities in disturbed series, with special reference to Wolfer's sunspot numbers," *Philosophical Transactions of the Royal Society of London*, ser. A, vol. 226, pp. 267–298, 1927.

[55] W. Gilbert, "On periodicity in series of related terms," in *Proceedings of the Royal Society of London*, ser. A, vol. 131, pp. 518–532, 1931.

[56] G. Raux and D. Halverson, "An empirical distribution approach to the application of differential geometric robustness analysis," in *IEEE International Sym-*

posium on Wireless Communication Systems 2007, 16-19 Oct. 2007, Trondheim, Norway, pp. 236–240.

[57] P. Fjelstad and I. Ginchev, "Volume, surface area, and the harmonic mean," *Math. Mag.* vol. 76, pp. 126–129, Feb. 2003.

[58] A. Gray, "The intuitive idea of area on a surface." §15.3 in *Modern Differential Geometry of Curves and Surfaces with Mathematica,* 2nd ed. Boca Raton, FL: CRC Press, pp. 351–353, 1997.

[59] E. W. Weisstein, "Surface area", MathWorld, Online available. http://mathworld.wolfram.com/SurfaceArea.html, Jan 2007.

[60] N. J. Higham, "Matrix Norms," §6.2 in *Accuracy and Stability of Numerical Algorithms*, Philadelphia: Soc. Industrial and Appl. Math., 1996.

[61] I. S. Gradshteyn, and I. M. Ryzhik, "Jacobian determinant," §14.313 in *Tables of Integrals, Series, and Products,* 6th ed. San Diego, CA: Academic Press, pp. 1068–1069, 2000.

[62] C. P. Simon, and L. E. Blume, *Mathematics for Economists*, New York: W. W. Norton, 1994.

[63] Ya-lun Chou, *Statistical Analysis*. University of California: Holt International, 1969.

[64] J. F. Kenney and E. S. Keeping, "Harmonic mean." §4.13 in *Mathematics of Statistics*, Pt. 1, 3rd ed. Princeton, NJ: Van Nostrand, pp. 57–58, 1962.

[65] E. W. Weisstein, "Harmonic mean", MathWorld, Online available. http://mathworld.wolfram.com/HarmonicMean.html, Jan 2007.

[66] L. Hoehn, and I. Niven,"Averages on the move." *Math. Mag.*, vol. 58, pp. 151–156, May 1985.

[67] I. Chavel, *Riemannian Geometry: A Modern Introduction*. Cambridge: Cambridge University Press 1994.

[68] V. Guillemin and A. Pollack, *Differential Topology*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[69] R. Millman and G. Parker, *Elements of Differential Geometry*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[70] H. Reckziegel, *Mathematical Models from the Collections of Universities and Museums*. Munich, Germany: Braunschweig, 1986.

[71] I. Singer and J. Thorpe, *Lecture Notes on Elementary Topology and Geometry*. New York: Springer-Verlag, 1996.

[72] E. W. Weisstein, "Ellipsoid", MathWorld, Online available. http://mathworld.wolfram.com/Ellipsoid.html, Jan 2007.

[73] J. Kenney and E. Keeping, *Relation between Mean, Median, and Mode*, vol. 1, 3rd ed. Princeton, NJ: Mathematics of Statistics, 1962.

[74] J. Kenney and E. Keeping, *Confidence Interval Charts*, vol. 1. Princeton, NJ: Mathematics of Statistics, 1962.

[75] A. Jain and W. Waller, "On the optimal number of features in the classification of multivariate Gaussian data," *Pattern Recognition*, vol. 10, pp. 365–374, Mar. 1978.

[76] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* New York: Dover, 1972.

[77] G. Borros and V. Moll, *Irresistible Integrals: Symbolics, Analysis and Experiments in the Evaluation of Integrals.* Cambridge: Cambridge University Press, 2004.

[78] S. Kobayashi and K. Nomizu, *Foundations of Differential Geometry*, vol 1 and 2. New York: Wiley,1963.

[79] R Bishop and S Goldberg, *Tensor Analysis in Manifolds.* New York: MacMillan, 1968.

[80] M. Abramowitz and I. A. Stegun, chap 17: Leibnitz Integrals, pp. 234–278 *Handbook of Mathematical Functions.* New York: Dover Publications, 1964.

[81] ETSI Technical Specification document 145 008 v4.16.0, *Digital Cellular Telecommunications System* (phase 2+); Radio subsystem link control (3GPP TS 45.008, version 4.16.0, release 4).

[82] J. Tebbit, A. Agarwal, M. Narang and V. Varma, *Methods, Devices and Systems Relating to Reselecting Cells in a Cellular Wireless Communications System*, U. S. Patent App. No. 11/340893.

[83] J. Tebbit, A. Agarwal, M. Narang and V. Varma, *World Intellectual Property Organization,* (Publication Number: WO/2007/087636).

APPENDIX A

LIST OF VARIABLES

L_p : error for the estimator

$\pm\delta$: variation in the entry of the covariance matrix

P : performance function

$f_X(x)$: PDF of the random variable x

x[.] : data set

θ : unknown parameter

$\widehat{\theta}$: estimator for detection theory

$p(x[n];\theta)$: probability density function of data x[n] parameterized by the unknown parameter θ

σ^2 : variance of the random variable

E_n : n-dimensional Euclidian space

y_i : coordinates in the n-dimensional Euclidian space

$\mathcal{H}_0$: binary hypothesis (null hypothesis)

$\mathcal{H}_1$: binary hypothesis (alternative hypothesis)

α : false alarm probability

β : detection probability

$\widehat{T}$: threshold value for the test statistic

Γ : log likelihood ratio

x^1 : parameter or local coordinate in differential geometry

T_m : set of all tangent vectors at m

$\overline{C}_i$: covariant vector field

$v^i w^j$: tensor product

δ^i_j : Kronecker delta tensor

g_{ij} : metric tensor

ds^2 : arc-length differential

$\widetilde{C}$: set containing all the local coordinates

C : covariance matrix

E[x] : mean of X

$\Lambda(y_1, y_2)$: Neyman-Pearson optimal detector

$\overrightarrow{X}$: tangent vector

J : Lagrange multiplier

$(D_{\overrightarrow{X}}h)\Big|_{Extreme}$: directional derivative

X, Y : random variable

K_n : curvature at a point

$\chi(M)$: Euler characteristic of M

ΔMSE : total amount of change in MSE

APPENDIX B

JACOBIAN

```
Clear All
Close  All
x = {x1, x2, x3, x4, x5}
f = {Cos[x5]*Cos[x4]*Cos[x3]Cos[x2]*Cos[x1],
    Cos[x5]*Cos[x4]*Cos[x3]Cos[x2]*Sin[x1],
    Cos[x5]*Cos[x4]*Cos[x3]Sin[x2],
    Cos[x5]*Cos[x4]*Sin[x3],
    Cos[x5]*Sin[x4]}
M = Outer[D, f, x]
FullSimplify[Det[M = Outer[D, f, x]]]


y = {y1, y2}
g = {Cos[y2]*Cos[y1], Cos[y2]*Sin[y1]}
N2 = Outer[D, g, y]
FullSimplify[Det[N2 = Outer[D, g, y]]]
```

VITA

Vishal Vinod Varma was born in India. He is the son of Mr. Vinod and Sunita Varma, husband of Ruchi Varma, and the brother of Ashish Varma. He graduated from Indian Institute of Technology, Bombay, India in August 1999 with a Bachelor of Technology in electrical engineering. He then worked for Technofour Ltd, where he was involved in designing products for Automated Vision and Inspection Systems. Vishal attended Texas A&M University where he obtained a M.S. in December 2002, in electrical engineering. After that, Vishal continued on to his pursue his doctoral degree at Texas A&M University. He received his Ph.D. in electrical engineering in December 2008. He developed framework for Robust System Architecture in Wireless Systems, with emphasis on residual stationary channels, under the supervision of Dr. Don Halverson. From 2004-05 Vishal worked at Qualcomm Inc. as System Integration Engineer focusing on Qualcomm's Mobile Station Modems (MSM) product family - chip-sets for Wide-band Code Division Multiple Access (W-CDMA) and General Packet Radio Service/Enhanced Data rates for GSM Evolution (GPRS/EDGE) Systems. His research interests are in Orthogonal Frequency Division Multiplexing (OFDM), Digital Signal Processing, Wireless Communication Systems, Broadband Wireless, Stochastic Financial Modeling. He is currently working as Technical Lead, with Intel Corp. at Mobile Wireless Division, focusing on IEEE 802.16 Standard.

Vishal can be reached at:

Intel Corporation, MS:SC12-512, 2200 Mission College Boulevard,

Santa Clara, CA 95054

Email: varma.vishal@gmail.com

The typist for this dissertation was Vishal Vinod Varma.